NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

AIRBORNE PASSIVE
TARGET MOTION ANALYSIS

by

Jay A. Gutzler

September 1987

Thesis Advisor H.A. Titus

Approved for public release; distribution is unlimited.

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) 62	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a NAME OF FUNDING / SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)

AIRBORNE PASSIVE TARGET MOTION ANALYSIS

12 PERSONAL AUTHOR(S)
GUTZLER, Jay A.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM ____ TO ____		

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB GROUP	Kalman Filter; Passive Target Motion Analysis; Airborne Tracking

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Kalman filtering techniques are applied to a two sensor bearings only passive target motion analysis problem. An algorithm is developed to simulate tracking long range maneuvering airborne targets. The target tracking performance of the filter is evaluated using computer generated noisy bearing measurements. The performance of the filter is satisfactory given reasonable initial conditions and measurement noise.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	UNCLASSIFIED

22a NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE (Include Area Code)	22c OFFICE SYMBOL
Prof H. A. Titus	(408) 646-2156	62 Ts

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

Approved for public release; distribution is unlimited.

Airborne Passive Target Motion Analysis

by

Jay A. Gutzler
Lieutenant, United States Navy
B.S., U.S. Naval Academy, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING
(ELECTRONIC WARFARE)

from the

NAVAL POSTGRADUATE SCHOOL
September 1987

Author: _____
Jay A. Gutzler

Approved by: _____
H.A. Titus, Thesis Advisor

_____
A. Gerba Jr., Second Reader

_____
J. Sternberg, Chairman,
Electronic Warfare Academic Group

_____
Kneale T. Marshall
Acting Academic Dean

ABSTRACT

Kalman filtering techniques are applied to a two sensor bearings only passive target motion analysis problem. An algorithm is developed to simulate tracking long range maneuvering airborne targets. The target tracking performance of the filter is evaluated using computer generated noisy bearing measurements. The performance of the filter is satisfactory given reasonable initial conditions and measurement noise.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

Air defense of a carrier battle group is becoming significantly more complex due not only to the increased speed and range of potentially hostile aircraft but also to more capable enemy targeting systems and greater cruise missile ranges. To reduce the probability of an aircraft carrier being successfully targeted by an enemy cruise missile carrying aircraft, it is imperative that fighter intercept be accomplished beyond the maximum range of the cruise missile. Long range over-the-horizon (OTH) target detection and tracking are necessary to achieve this goal.

A major obstacle common to all air defense scenarios is the enemy's use of electronic countermeasures (ECM). Attacking enemy aircraft will undoubtedly employ jamming as well as other forms of ECM to degrade or deny effective tracking by active systems. Therefore, the ability to *passively* track is required in order to successfully engage attacking aircraft in a dense ECM environment.

One viable approach to this problem is passive Target Motion Analysis (TMA). The purpose of TMA is to determine the target's position, course and speed through a series of passive noisy measurements. For the air defense scenario, these passive measurements may be lines of bearing (LOB) obtained from the enemy aircraft's jamming strobes or from the electromagnetic radiation of the aircraft's long range targeting radar. (Theses),

Passive bearings only TMA may be performed by one or more sensors. The two primary considerations in evaluating TMA performance are solution accuracy and timeliness. Single sensor TMA requires that the observer aircraft perform zig zag maneuvers to establish a target bearing rate so that the range to the target may be estimated. One drawback to single sensor TMA is the fact that these maneuvers may detract from the observer aircraft's primary mission. Also, a reasonable initial estimate of the target's state (position, course and speed) is necessary to ensure that the tracking solution converges in a timely manner, if indeed it converges at all. An inherent difficulty with bearings only TMA by a single sensor is that the solution accuracy and timeliness rely quite heavily upon a "good" a priori estimate of target range. Consequently, in a long range tracking scenario where the range to the target may exceed several hundred miles, accurate tracking by a single sensor using only bearing observations is extremely arduous and rather impractical.

A practical solution to long range OTH passive tracking is multi-sensor triangulation. High speed air targets can be accurately tracked by two or more highly directional sensors that are spaced sufficiently far apart. The primary reason that multi-sensor tracking is superior to single sensor TMA is that estimates of target range are continually being generated through triangulation of sensor bearing lines. Multi-sensor tracking is far less dependent on accurate a priori state estimates than is single sensor tracking for timely convergence. The major obstacle, however, in using the multi-sensor triangulation method is a practical one: very close cooperation is required between the observers in order to achieve an accurate tracking solution. Three ingredients are required to localize a target: the position of each sensor, the time of the observation, and the bearing measurement from each sensor. Ideally, all observations would be performed synchronously. If asynchronous lines of bearing are encountered, then computer processing is required to interpolate these LOB's to produce "synchronous" measurements. The observers are, in effect, remote sensors that transmit noisy bearing data to a central processing platform where the actual target tracking is performed. For tracking a long range and rapidly closing air target, triangulation provides a significantly more accurate and timely tracking solution.

Because each sensor generates its own sequence of noisy bearing observations, the Kalman filter is ideally suited for determining a target's position and motion. This thesis investigates the two sensor bearings only tracking problem in a computer simulation that employs Kalman filtering techniques. The simulation generates the target and observer tracks as well as noisy bearing measurements from each sensor to the target. The noisy bearings are then processed by the Kalman filter to provide continual estimates of the target's state. The tracking algorithm is used in several scenarios to determine the effect various sensor bearing accuracies and initial estimates have on the filter's performance.

The aim of this research is to examine how well the filter performs in tracking a non-maneuvering target before investigating the more difficult case of a maneuvering target. The problem geometry will be presented first, followed by the development of the system and measurement models. Relevant equations from Kalman filtering theory will then be briefly reviewed before the actual tracking algorithm is analyzed in detail. The results of several simulation runs using various parameters will be examined. Also, the effect of target maneuvers on filter stability will be assessed. The final chapter will summarize the results of this research and will present conclusions and recommendations for further study.

II. PROBLEM DESCRIPTION

A. INTRODUCTION

As the name implies, over-the-horizon detection and tracking means positioning sensors out near the radar horizon to look over the edge and pass their observations back to a central data fusion point for analysis. This data fusion point need not be a surface combatant; it could be a command and control aircraft. The use of an airborne command and control platform extends the range to which an air target can be effectively tracked. The basic idea behind OTH tracking is to place the remote sensors far enough apart so that effective triangulation fixes may be taken but not so far apart that they are beyond the range at which they can communicate with the central processing platform. The command and control aircraft should ideally be positioned on the threat axis between the incoming air attack and the high value unit (HVU) that is being protected. Figure 2.1 depicts the general geometry of a basic OTH detection and tracking scenario.

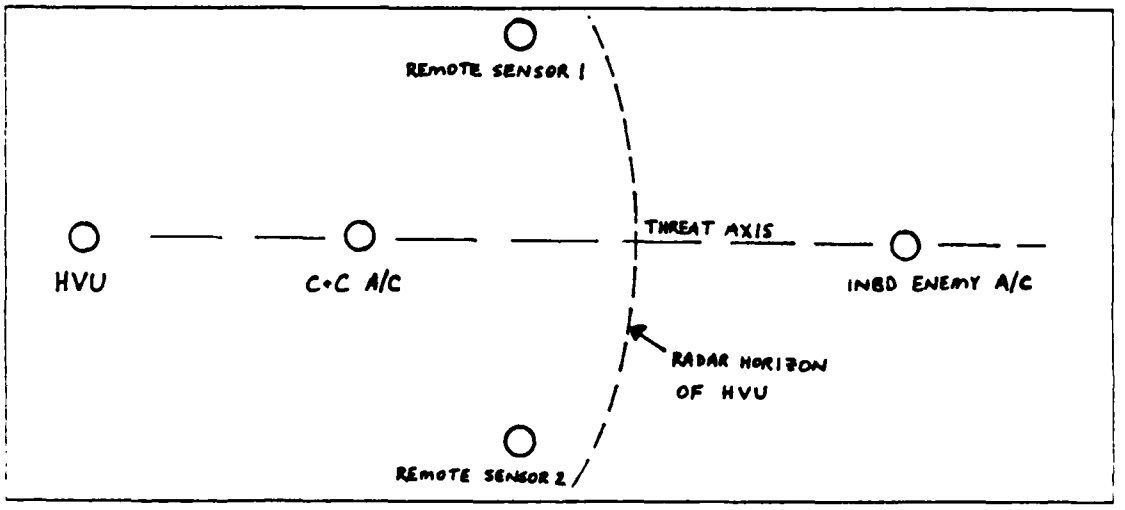

Figure 2.1 Basic Over-the-Horizon Detection and Tracking Scenario.

In this chapter the geometry of the two sensor TMA problem will be presented along with a development of the target motion and noise-free measurement equations.

B. PROBLEM GEOMETRY

Consider the target-observer geometry in the two dimensional plane as shown in

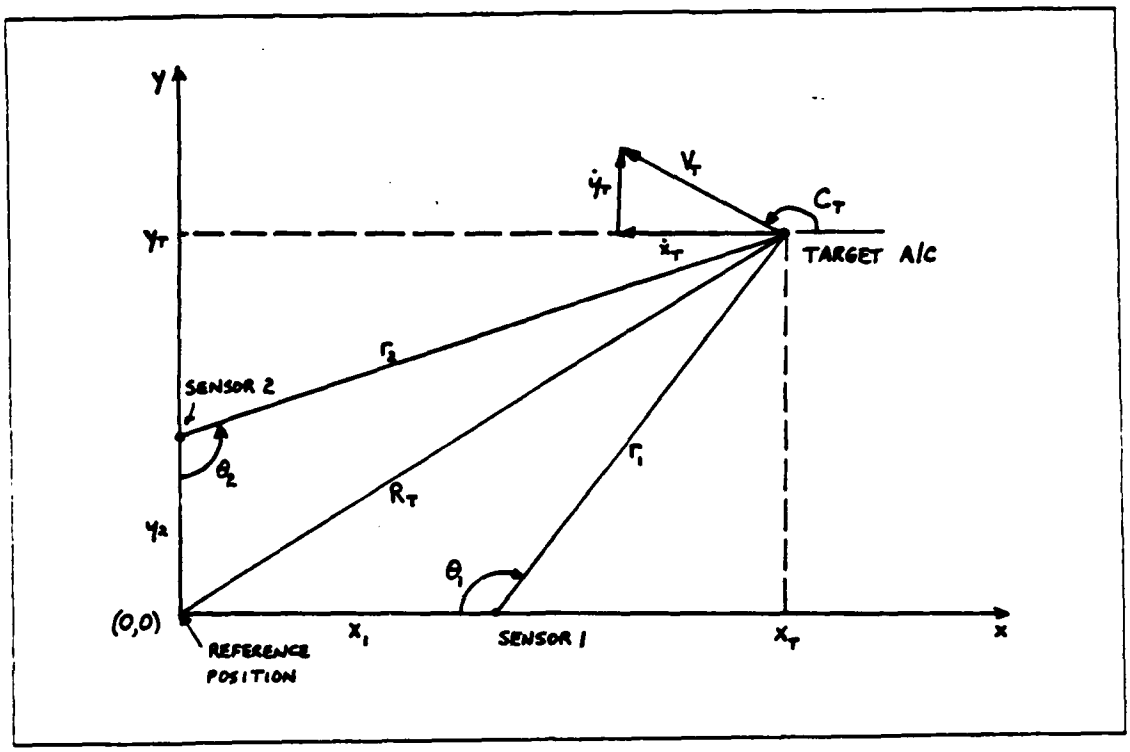

Figure 2.2 Target-Observer Geometry.

Figure 2.2. The target is located at (x_T , y_T) from a defined reference position. The origin may be defined as either a fixed latitude/longitude coordinate or the position of a high value unit such as an aircraft carrier (whose position is relatively stationary). The x and y components of target velocity are denoted as $\dot{x}_T$ and $\dot{y}_T$ and are the Cartesian equivalents of the target's course and speed, C_T and V_T. Sensor 1 is located at (x_1 , 0) and is only able to move along the x-axis with velocity $\dot{x}_1$. Likewise, sensor 2 is located at (0 , y_2) and is only able to move along the y-axis with velocity $\dot{y}_2$. As shown in Figure 2.2, the bearing from sensor 1 to the target is θ_1 and the bearing from sensor 2 to the target is θ_2. The ranges to the target from sensors 1 and 2 are denoted as r_1 and r_2 respectively, and the range of the target from the origin is denoted R_T.

1. Problem Assumptions

The following assumptions are made concerning the problem:

1. The target is initially inbound and remains within the first quadrant.
2. The target maintains a constant speed but is free to change course.

3. Both sensor positions are known precisely.
4. Bearing data from each sensor are continuously observed and are received synchronously.
5. Bearing noise is zero mean and Gaussian with variances σ_1^2 and σ_2^2 for sensors 1 and 2, respectively.
6. Target turns are modeled as instantaneous (i.e., no turn radius).
7. Target and sensor altitudes have negligible effect at long ranges.

The last assumption is entirely reasonable since the difference between the target's slant range and two dimensional range is less than a fraction of 1% for ranges exceeding 300 nautical miles.

2. **Practical Geometrical Considerations**

Placing the airborne sensors on orthogonal axes is chosen not only because it simplifies the geometry but also because it provides adequate sensor separation with which to perform accurate triangulation. Ideally, the most accurate triangulation fix is formed from the intersection of two perpendicular lines of bearing. Perpendicular LOB's in real world scenarios, however, are extremely difficult to obtain for a number of reasons. One reason is that the maximum range and on station time for airborne sensors are limited. Also, if electromagnetic energy from the target is being used to obtain LOB's, it is important that both sensors be positioned within the main sector beam pattern. Figure 2.3 illustrates some of these considerations. In Figure 2.3 (a), it can be seen that in the attempt to obtain perpendicular LOB's to the target, sensor 1 is beyond the radar horizon of the command and control aircraft and is thus unable to pass any bearing observations. Figure 2.3 (b) shows both sensors lying within the sector scan limits of the target's surveillance radar. While it is not necessary for both sensors to be within the main beam simultaneously, both must be able to detect the beam's presence within a reasonable time period, a factor which depends on the radar's scan rate.

The scenario where the sensors are positioned on orthogonal axes could easily be modified to the more general case where the sensors are located on radials that are separated, say, by 60 degrees. Since the sensors are now closer together, range estimates to the target would be somewhat degraded. Also, by adding a third sensor on a radial 120 degrees from the first sensor and 60 degrees from the second would provide a wider sector coverage as well as improved tracking accuracy.

The simulations that have been run in this thesis involve extreme ranges from each sensor to the target. It has been assumed throughout that the target and sensor

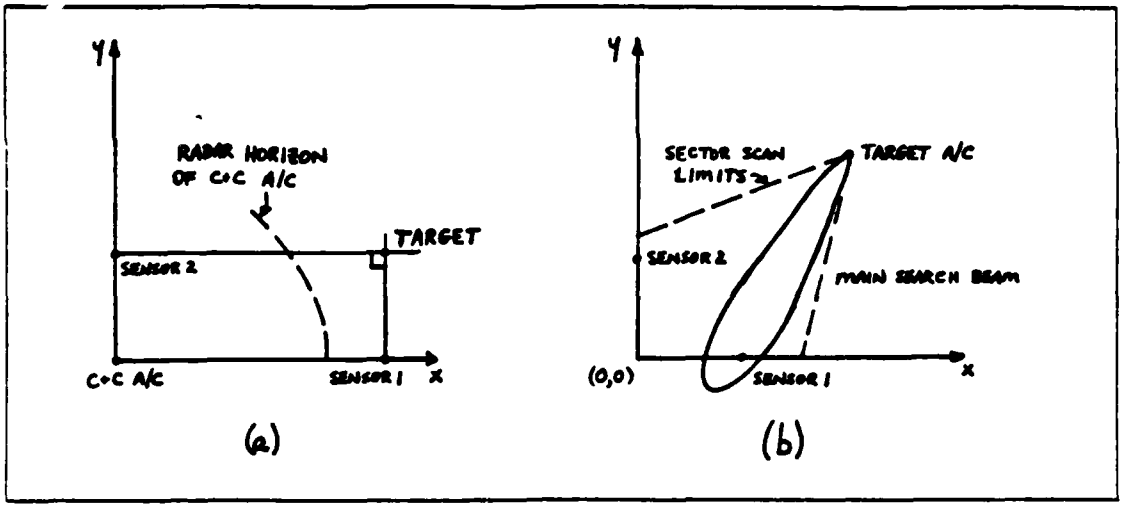

Figure 2.3 Practical Geometric Considerations.

aircraft are flying at medium to high altitudes so that all observations will meet radar horizon range constraints. It should be noted, however, that the practical limiting factors for maximum detection range are the strength and radio frequency (RF) of the intercepted source signal. Also, bearing accuracies depend on the RF of the signal. Extreme detection ranges, sometimes between 400 and 500 nautical miles, have been used to represent a worst case tracking problem; shorter ranges would yield a more accurate tracking solution.

C. SYSTEM MODEL

As shown in Figure 2.2, lines of bearing from two airborne sensors are used to determine the target's state (position, course and speed). Using a Cartesian coordinate system, a four dimensional state vector, x_T, is chosen.

$$x_T = \begin{bmatrix} x_T \\ \dot{x}_T \\ y_T \\ \dot{y}_T \end{bmatrix} \qquad \text{(eqn 2.1)}$$

It should be noted that the system model is in no way limited to a Cartesian reference frame or state vector; the Cartesian coordinate system was chosen merely for its mathematical simplicity.

1. **Target Maneuvers**

Two basic scenarios are addressed in this thesis. The first one involves tracking a non-maneuvering target and the second involves a maneuvering target. In both cases the target is assumed to be initially inbound and any target maneuver will consist of the target changing only its course and maintaining its speed. Figure 2.4 shows the target tracks that will be examined in subsequent chapters.

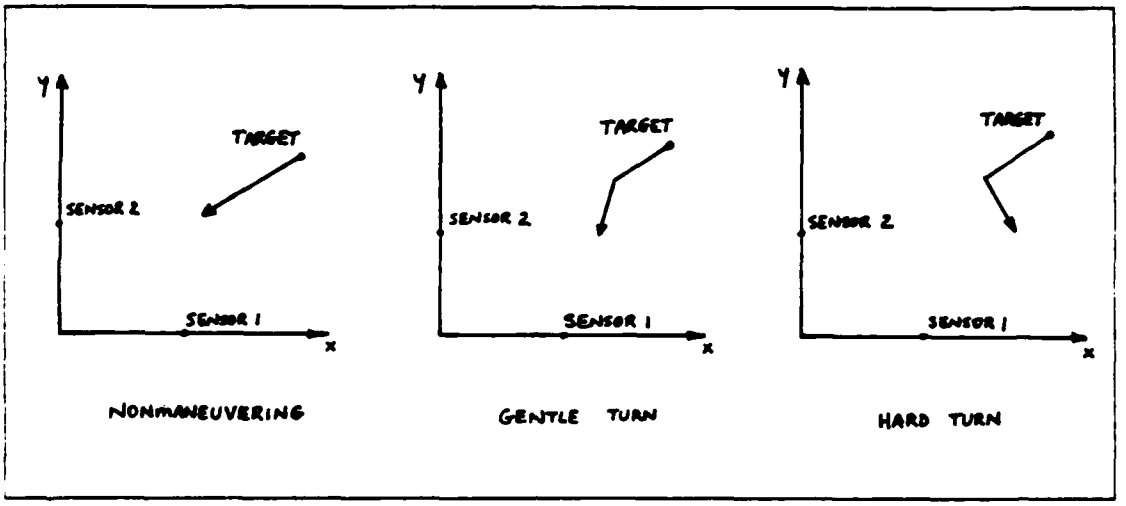

Figure 2.4 Representative Target Tracks.

It is assumed that target maneuvers can be modeled by using white random forcing functions. As shown in Figure 2.5, target maneuvers may be thought of as acceleration along its course (radial acceleration) and acceleration perpendicular to its course (turn rate). Let the random variables $\delta_{\dot{v}}$ and $\delta_{\dot{\theta}}$ denote the target's acceleration along its course and acceleration perpendicular to its course, respectively. Both $\delta_{\dot{v}}$ and $\delta_{\dot{\theta}}$ denote random changes of the target and are assumed to be independent and zero mean with variances $\sigma_{\dot{v}}^2$ and $\sigma_{\dot{\theta}}^2$. Because of the extremely long ranges involved in the simulations, target maneuvers have been modeled as instantaneous changes of position according to the time interval used. That is, the simulation enables the target to turn 90° in two seconds. While it is acknowledged that this kind of turn rate is quite artificial, it is informative to see what effect such a drastic turn rate has on tracking performance and stability. The variances used in subsequent scenarios are:

$$\sigma_{\dot{\zeta}}^2 = (300\text{knots/sec})^2 \qquad \text{(eqn 2.2)}$$

$$\sigma_{\dot{\theta}}^2 = (45\ \text{deg/sec})^2 \qquad \text{(eqn 2.3)}$$

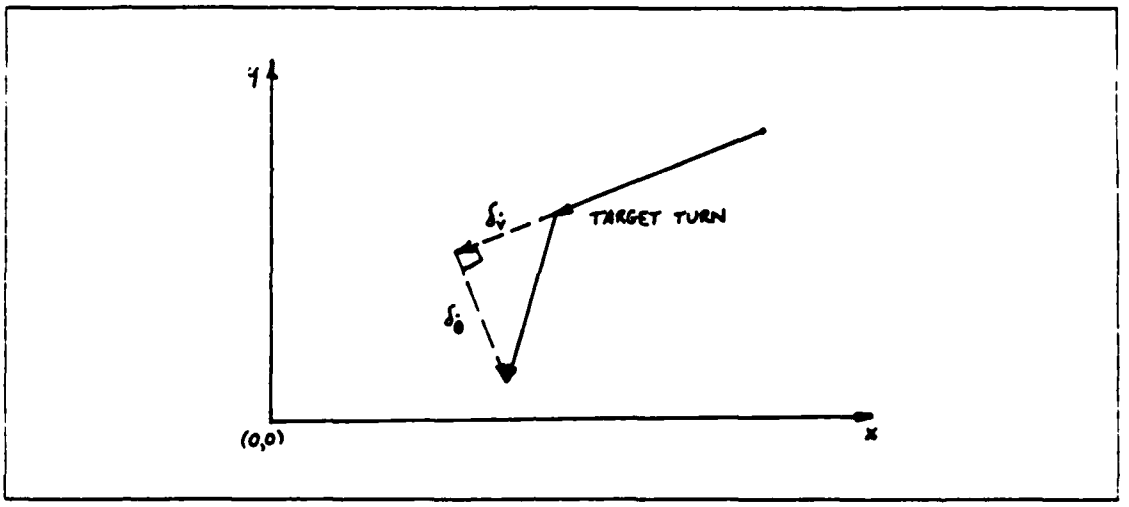

Figure 2.5 Geometry of Target Maneuvers.

a. Equations of Motion

Let T represent the time interval between observations. If k represents the k^{th} observation and t_k the discrete time of the k^{th} observation, then T may be expressed as

$$T = t_k - t_{k-1} \qquad \text{(eqn 2.4)}$$

Referring to Figure 2.2 and Figure 2.5, target motion may be described by the difference equations:

$$\underline{x}_T(k+1) = \begin{bmatrix} x_T(k+1) \\ x_T(k+1) \\ y_T(k+1) \\ y_T(k+1) \end{bmatrix} = \begin{bmatrix} x_T(k) + T\,x_T(k) + f_1(\delta_v, \delta_\theta, k) \\ x_T(k) + f_2(\delta_v, \delta_\theta, k) \\ y_T(k) + T\,y_T(k) + f_3(\delta_v, \delta_\theta, k) \\ y_T(k) + f_4(\delta_v, \delta_\theta, k) \end{bmatrix} \qquad \text{(eqn 2.5)}$$

The random forcing functions f_1 through f_4 are included to account for random changes in speed and heading which occur for a moving target. Equation 2.5 may be written in matrix form as

$$x(k+1) = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_T(k) \\ x_T(k) \\ y_T(k) \\ y_T(k) \end{bmatrix} \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \begin{bmatrix} x_T(k) \\ y_T(k) \end{bmatrix} \quad \text{(eqn 2.6)}$$

or more concisely as

$$x_{k+1} = \Phi_k x_k + \Gamma_k w_k \quad \text{(eqn 2.7)}$$

where x_k is the 4×1 state vector
Φ_k is the 4×4state transition matrix
w_k is the 2×1 vector of random forcing functions
Γ_k is the 4×2 state forcing matrix.

The terms of the random forcing function vector w_k represent the accelerations in the x and y directions caused by target maneuvers. The state forcing matrix Γ_k represents a uniform constant acceleration model of target motion. If the time interval T between measurements is assumed constant, then Φ_k may be replaced by a constant state transition matrix Φ and Γ_k may be replaced by a constant state forcing matrix Γ. Revising Equation 2.7, the linear system model can be expressed as

$$x_{k+1} = \Phi x_k + \Gamma w_k \quad \text{(eqn 2.8)}$$

D. NOISE-FREE MEASUREMENT EQUATION

As illustrated in Figure 2.2, the positions of sensors 1 and 2 along with their respective bearings to the target, θ_1 and θ_2, uniquely define the target's position (X_T, Y_T). The target's position from noise-free bearing observations may be expressed as

$$x_T = \frac{(y_2 \cos\theta_1 - x_1 \sin\theta_1) \sin\theta_2}{\cos(\theta_1 + \theta_2)} \quad \text{(eqn 2.9)}$$

$$y_r = \frac{(x_1 \cos \theta_2 - y_2 \sin \theta_2) \sin \theta_1}{\cos(\theta_1 + \theta_2)} \qquad \text{(eqn 2.10)}$$

The positions and speeds for the airborne sensors may be chosen arbitrarily for input into the tracking algorithm. Each sensor's position is assumed to be known precisely for each time interval. The sensors may both head inbound or outbound or they may go in alternate directions. Care must be exercised in choosing sensor positions and speeds so as to avoid having lines of bearing that are collinear (each sensor is pointing at the other). What results in this case is an extremely thin and elongated error ellipse which momentarily degrades tracking accuracy at the moment that the lines of bearing are coincident.

It should be noted that using two sensors eliminates the need for any extraneous observer maneuvering as is the case for a single sensor. The observer aircraft can basically fly straight and level and collect more reliable bearings to the target. Also, the sensor's position is known more precisely since it is not decelerating and accelerating into and out of turns.

III. KALMAN FILTERING

A. INTRODUCTION

The technique of Kalman filtering is ideally suited to the problem of passive tracking. The following sections briefly describe the theory and results of Kalman filtering and how it is applied to the long range airborne TMA problem. For a more in-depth development of the Kalman filter, the reader is referred to [Refs. 1,2].

B. THE KALMAN FILTER

The purpose of the Kalman filter is to keep track of the state of a system through a sequence of noisy measurements. This is accomplished by recursively updating an estimate of the state by processing a sequence of noisy observations in such a manner as to reduce as much as possible the effect of measurement errors.

The Kalman filter is a predictor-corrector type estimator that propagates an estimate, $\hat{\mathbf{x}}$, of the target state along with an associated covariance matrix, $\mathbf{P}$, which reflects the degree of confidence placed in the accuracy of the state estimate. The Kalman filter is carried out in two alternating stages. First, previous estimates of $\mathbf{x}$ and $\mathbf{P}$ are extrapolated one time step ahead based on the assumed system dynamics; this is referred to as the Movement Step. These extrapolated values are then used to compute a set of optimum weights called Kalman gains. The gains are applied to the prediction and to a new observation in a Measurement Step, which provides an updated estimate of the state and its covariance. This process is then repeated. [Ref. 3]

1. Assumptions

The following assumptions are made:

1. The random forcing function $\mathbf{w}_k$ is zero mean and uncorrelated with covariance $\mathbf{Q}_k$.
2. The measurement noise $\mathbf{v}_k$ is zero mean and is correlated with covariance $\mathbf{R}_k$.
3. The random forcing function $\mathbf{w}_k$ and measurement noise $\mathbf{v}_k$ are uncorrelated.
4. The initial state $\mathbf{x}_0$ is a random variable with known mean $\hat{\mathbf{x}}_{0|-1}$ and covariance $\mathbf{P}_{0|-1}$.

2. Definitions

1. The estimated state vector *after* k observations is denoted by $\hat{x}_{k|k}$ and the predicted state vector *before* the k^{th} observation is represented by $\hat{x}_{k|k-1}$.

2. The state estimation error vector ε_k is defined as the difference between the estimated state and the true state

$$\varepsilon_k \equiv \hat{x}_{k|k} - x_k \qquad \text{(eqn 3.1)}$$

and the predicted state estimation vector $\varepsilon_{k\ k-1}$ is defined as the difference between the predicted state and the true state

$$\varepsilon_{k|k-1} \equiv \hat{x}_{k|k-1} - x_k \qquad \text{(eqn 3.2)}$$

3. The covariance of estimation error matrix $P_{k|k}$ is defined as

$$P_{k|k} = E\{ \varepsilon_k \varepsilon_k^T \} \qquad \text{(eqn 3.3)}$$

and the predicted covariance of state error matrix $P_{k|k-1}$ is defined as

$$P_{k|k-1} = E\{ \varepsilon_{k|k-1} \varepsilon_{k|k-1}^T \} \qquad \text{(eqn 3.4)}$$

4. The state excitation covariance matrix is given by

$$Q_k = E\{ \Gamma w_k w_k^T \Gamma^T \} \qquad \text{(eqn 3.5)}$$

5. The Kalman filter is an optimal estimator that minimizes the sum of the variances of the estimation error, i.e.,

$$E\{\varepsilon_1(k)^2\} + E\{\varepsilon_2(k)^2\} + \ldots + E\{\varepsilon_n(k)^2\} \qquad \text{(eqn 3.6)}$$

Enter known matrices and a priori estimates:

$$\hat{\mathbf{x}}_{(0|-1)},\ \mathbf{P}_{(0|-1)},\ \mathbf{R}_{(0)},\ \mathbf{H},\ \boldsymbol{\Phi}$$

Compute the Kalman gain:

$$\mathbf{G}_{(k)} = \mathbf{P}_{(k|k-1)}\mathbf{H}^T\{\ \mathbf{H}\mathbf{P}_{(k|k-1)}\mathbf{H}^T + \mathbf{R}_{(k)}\ \}^{-1}$$

MEASUREMENT STEP

$$\hat{\mathbf{x}}_{(k|k)} = \hat{\mathbf{x}}_{(k|k-1)} + \mathbf{G}_{(k)}\{\ \mathbf{z}_{(k)} - \mathbf{H}\hat{\mathbf{x}}_{(k|k-1)}\ \}$$

$$\mathbf{P}_{(k|k)} = \{\ \mathbf{I} - \mathbf{G}_{(k)}\mathbf{H}\ \}\mathbf{P}_{(k|k-1)}$$

MOVEMENT STEP

$$\hat{\mathbf{x}}_{(k+1|k)} = \boldsymbol{\Phi}\,\hat{\mathbf{x}}_{(k|k)}$$

$$\mathbf{P}_{(k+1|k)} = \boldsymbol{\Phi}\mathbf{P}_{(k|k)}\boldsymbol{\Phi}^T + \mathbf{Q}_{(k)}$$

Compute $\mathbf{R}_{(k)}$ and $\mathbf{Q}_{(k)}$

Increment k by 1

Figure 3.1 Kalman Filter Algorithm.

3. Kalman Filter Algorithm

Figure 3.1 summarizes the discrete Kalman filter algorithm. For the particular TMA problem presented in this thesis, the 2×4 measurement matrix $\mathbf{H}_k$ and the 4×4 state transition matrix $\mathbf{\Phi}_k$ are both known, constant matrices and may be represented by $\mathbf{H}$ and $\mathbf{\Phi}$. An a priori estimate $\hat{\mathbf{x}}_{0|-1}$ of the target's state with an associated initial error covariance matrix $\mathbf{P}_{0|-1}$, as well as an initial estimate of the measurement noise covariance matrix $\mathbf{R}_0$ must be input into the filter algorithm. The algorithm computes the Kalman gain $\mathbf{G}_k$ based on these a priori values and then updates the estimate of the target's state when it receives a measurement. The error covariance matrix is also updated. Next, the state estimate and its error covariance matrix are projected one time step ahead based on the assumed system dynamics. The measurement noise covariance $\mathbf{R}_k$ and the state excitation covariance matrix $\mathbf{Q}_k$ are then computed before k is incremented by one and the whole process is repeated.

C. FUNCTIONS, MATRICES, AND EQUATIONS

In this section, the Kalman filter algorithm will be applied to the long range passive airborne tracking problem. A brief derivation of the random forcing function w_k, the state excitation covariance matrix $\mathbf{Q}_k$, the measurement equation $\mathbf{z}_k$, and the measurement noise covariance matrix $\mathbf{R}_k$ is given next.

1. Random Forcing Function

Recalling equation (2-6), the two dimensional random forcing function w_k represents the acceleration in the x and y directions caused by target maneuvers.

$$\underline{w}_k = \begin{bmatrix} \ddot{x}_k \\ \ddot{y}_k \end{bmatrix} = \begin{bmatrix} \dot{y}_k \delta_{\dot{\theta}} + (\dot{x}_k / v_k)\delta_{\dot{v}} \\ -\dot{x}_k \delta_{\dot{\theta}} + (\dot{y}_k / v_k)\delta_{\dot{v}} \end{bmatrix} \qquad \text{(eqn 3.7)}$$

where $v_k = (\dot{x}_k^2 + \dot{y}_k^2)^{1/2}$.

Since the random variables $\delta_{\dot{v}}$ and $\delta_{\dot{\theta}}$ were assumed to be zero mean, it follows that the random forcing function w_k is also zero mean. The variances of the x and y accelerations, denoted by $\sigma_{\ddot{x}}^2$ and $\sigma_{\ddot{y}}^2$ respectively, are

$$\sigma_{\ddot{x}}^2 = E[\ddot{x}_k] = \dot{y}_k^2 \sigma_{\dot{\theta}}^2 + \left(\frac{\dot{x}_k}{v_k}\right)^2 \sigma_{\dot{v}}^2 \qquad \text{(eqn 3.8)}$$

$$\sigma_{\ddot{y}}^2 = E[\ddot{y}_k] = \dot{x}_k^2 \sigma_{\dot{\theta}}^2 + \left(\frac{\dot{y}_k}{v_k}\right)^2 \sigma_{\dot{v}}^2 \qquad \text{(eqn 3.9)}$$

The covariance of the x and y acceleration $\sigma_{\ddot{x}\ddot{y}}^2$ is

$$\sigma_{\ddot{x}\ddot{y}}^2 = E\left[\ddot{x}_k \ddot{y}_k\right] = -\dot{x}_k \dot{y}_k \sigma_{\dot{\theta}}^2 + \frac{\dot{x}_k \dot{y}_k}{v_k} \sigma_{\dot{v}}^2 \qquad \text{(eqn 3.10)}$$

Therefore, the random forcing function covariance matrix Q_k' is

$$Q_k' = E\{ w_k w_k^T \} = \begin{bmatrix} \sigma_{\ddot{x}}^2 & \sigma_{\ddot{x}\ddot{y}}^2 \\ \sigma_{\ddot{x}\ddot{y}}^2 & \sigma_{\ddot{y}}^2 \end{bmatrix} \qquad \text{(eqn 3.11)}$$

where $\sigma_{\ddot{x}}^2$, $\sigma_{\ddot{y}}^2$, and $\sigma_{\ddot{x}\ddot{y}}^2$ are computed at the predicted values of x_T and y_T.

2. **State Excitation Covariance Matrix**

The purpose of the state excitation covariance matrix Q_k is to account for model inaccuracies or for a target that has maneuvered. It is basically a "procedure for masking the effects of modeling errors" [Ref. 2: p. 163]. In effect, the state excitation covariance matrix increases the size of the predicted covariance of error matrix which in turn increases the filter gains. As more observations are processed, Q_k prevents the Kalman gains from approaching zero by continually injecting uncertainty into the predicted state estimate at each iteration. A nonzero Q_k slightly degrades the filter's accuracy when the target is not maneuvering but it helps prevent filter divergence when the target does maneuver. As stated in equation 3.5, the state excitation covariance matrix is

$$Q_k = \Gamma Q_k' \Gamma^T = \begin{bmatrix} \frac{T^4}{4}\sigma_{\ddot{x}}^2 & \frac{T^3}{2}\sigma_{\ddot{x}}^2 & \frac{T^4}{4}\sigma_{\ddot{x}\ddot{y}}^2 & \frac{T^3}{2}\sigma_{\ddot{x}\ddot{y}}^2 \\ & T^2\sigma_{\ddot{x}}^2 & \frac{T^3}{2}\sigma_{\ddot{x}\ddot{y}}^2 & T^2\sigma_{\ddot{x}\ddot{y}}^2 \\ & & \frac{T^4}{4}\sigma_{\ddot{y}}^2 & \frac{T^3}{2}\sigma_{\ddot{y}}^2 \\ \text{SYMMETRIC} & & & T^2\sigma_{\ddot{y}}^2 \end{bmatrix} \qquad \text{(eqn 3.12)}$$

3. **Measurement Equation**

In this TMA problem, the observations are noisy (x,y) positions. It is the intersection of noisy lines of bearing that form the noisy (x,y) position of the target that is input into the Kalman filter algorithm. Because the observations are of the same form as the state vector, the measurement equation is linear and is expressed as

$$z_k = H x_k + v_k \qquad \text{(eqn 3.13)}$$

where z_k is the 2×1 measurement vector

H is the 2×4 measurement matrix

v_k is the 2×1 measurement noise vector

x_k is the 4×1 state vector.

The equation may be written explicitly as

$$\begin{bmatrix} z_x \\ z_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ \dot{x}_k \\ y_k \\ \dot{y}_k \end{bmatrix} + \begin{bmatrix} v_{1k} \\ v_{2k} \end{bmatrix} \qquad \text{(eqn 3.14)}$$

The most important part of the measurement equation is an accurate description of the measurement noise vector v_k The measurement noise vector expresses the statistical nature of the noisy (x,y) position that is derived from the intersection of two noisy lines of bearing. These bearing measurement errors are assumed to be independent and zero mean with variances σ_1^2 and σ_2^2, for sensor 1 and sensor 2 respectively.

It is important to note that the bearing errors between sensors are statistically uncorrelated; one sensor's bearing accuracy has nothing to do with any other sensor's bearing accuracy. However, in describing the resulting intersection in Cartesian coordinates, the noisy x and noisy y positions are correlated. The only case where the noisy x and noisy y positions are uncorrelated is when the lines of bearing are perpendicular.

4. **Error Ellipses**

An intuitive way to visualize the measurement equation is through the concept of error ellipses. Error ellipses give a geometric picture of the region around a noisy position or estimate where the true value is considered to lie. Figure 3.2 shows a bivariate Gaussian probability density function formed by the intersection of two lines of bearing with independent Gaussian distributions.

As can be seen in Figure 3.2, the lines of bearing intersect at an oblique angle, forming an asymmetric hump. While the bivariate Gaussian probability density function gives an interesting three dimensional depiction of two normally distributed bearing errors, it does not provide the information that is really needed, quickly. What

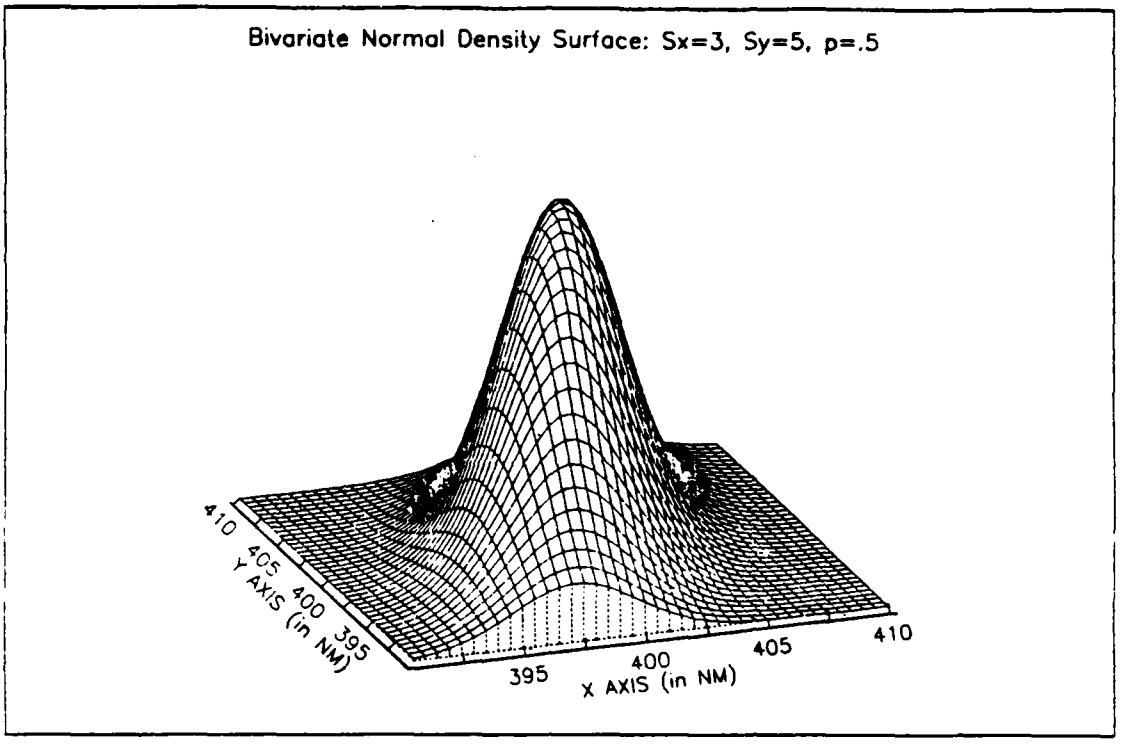

Figure 3.2 Bivariate Gaussian Probability Density Function.

is needed is an accurate picture of the measurement (or estimation) error. This uncertainty is best expressed geometrically by the error ellipse. The term "error ellipse" refers to the two dimensional surface of constant probability density. Figure 3.3 presents these error ellipses as contour lines of the bivariate Gaussian probability density function shown in Figure 3.2.

The various ellipse sizes in Figure 3.3 correspond to different constant probabilities. The fact that the ellipses are also rotated implies that the uncertainty in measurement error is indeed correlated with respect to x and y. The actual probabilities within a specified error ellipse may be computed through lengthy integration of the bivariate Gaussian probability density function over the surface of the ellipse. Some computed probabilities of the true value lying within the 1σ, 2σ, and 3σ error ellipses are .394, .865, and .989 respectively [Ref. 4: pp. 4-49].

Error ellipses are extremely useful in examining position error. Matrices containing the x and y position terms convey analytically what error ellipses display graphically. A 2×2 error covariance matrix which contains position components x and y is able to completely describe an ellipse. The main diagonal terms represent the

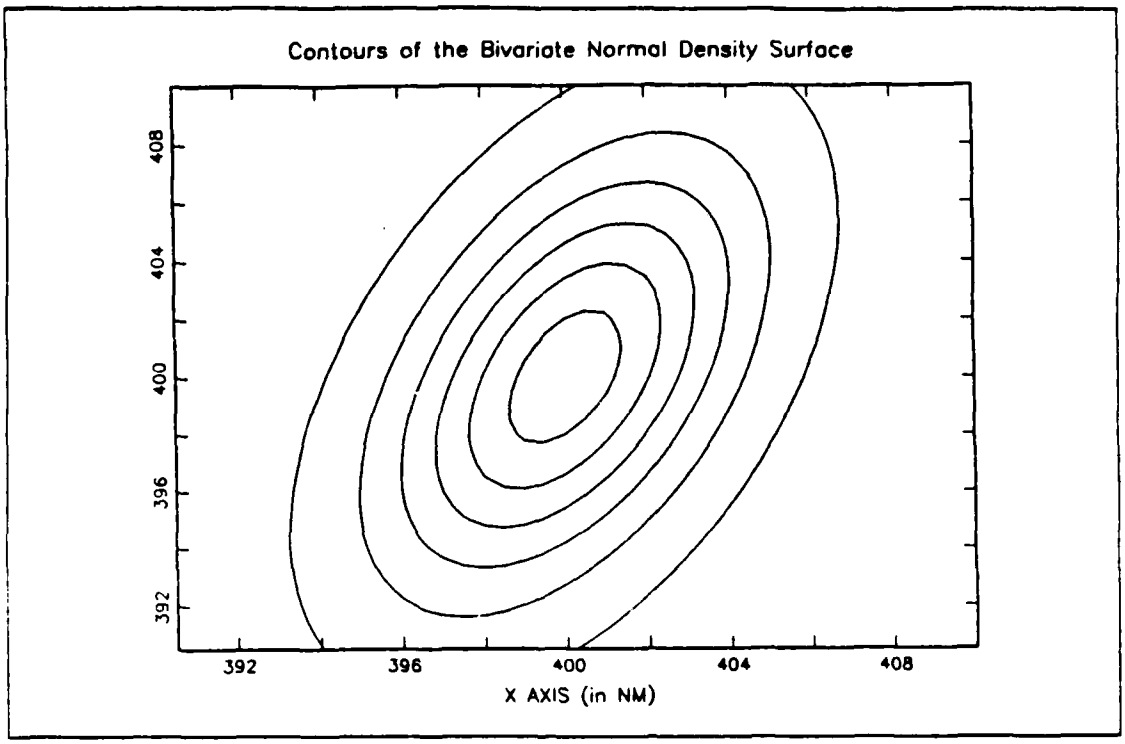

Figure 3.3 Error Ellipses as Contour Lines.

variances of the x and y positions respectively. The off diagonal terms represent the degree of x-y coupling and the orientation of the error ellipse in the x-y plane.

5. **Covariance of Measurement Error Matrix**

The covariance of measurement error matrix $\mathbf{R}_k$ uses the concept of error ellipses to accurately describe the noisiness and degree of coupling of (x,y) measurements obtained from intersections of noisy LOB's. The terms of the covariance of measurement error matrix $\mathbf{R}_k$ depend on the standard deviations of bearing error σ_1 and σ_2 of sensors 1 and 2 as well as the angle at which the lines of bearing intercept. The covariance of measurement error may be expressed as

$$R = \begin{bmatrix} \dfrac{\sigma_1^2 \sin^2\theta_2 + \sigma_2^2 \cos^2\theta_1}{\cos^2(\theta_1+\theta_2)} & \dfrac{-\sigma_1^2 \sin\theta_2 \cos\theta_2 - \sigma_2^2 \cos\theta_1 \sin\theta_1}{\cos^2(\theta_1+\theta_2)} \\ \dfrac{-\sigma_1^2 \sin\theta_2 \cos\theta_2 - \sigma_2^2 \cos\theta_1 \sin\theta_1}{\cos^2(\theta_1+\theta_2)} & \dfrac{\sigma_1^2 \cos^2\theta_2 + \sigma_2^2 \sin^2\theta_1}{\cos^2(\theta_1+\theta_2)} \end{bmatrix} \quad \text{(eqn 3-15)}$$

where θ_1 and θ_2 denote noisy bearing observations from sensors 1 and 2, respectively. The subscript k has been intentionally deleted from equation (3-15) only for ease of notation. At each discrete time interval t_k, new values of θ_1 and θ_2 are generated with which to compute the new measurement error covariance matrix, $\mathbf{R}_k$ A complete derivation of equation (3-15) is given in Appendix A for the interested reader.

IV. THE ALGORITHM

A. INTRODUCTION

This section discusses the development of the tracking algorithm. The algorithm is designed to simulate tracking a long range inbound enemy air target by triangulating noisy bearing observations from two airborne sensors. We want to be able to track a non-maneuvering target within a one percent range error. For a maneuvering target, we desire a stable filter response which quickly converges to the target's new state. The effect of various sensor bearing errors, a priori state estimates and initial error covariance matrices on filter accuracy and convergence time are investigated. Also, the effect of target maneuvers on filter stability is analyzed. Basically, the algorithm performs three functions:

1. The target and sensor tracks are generated.
2. Noisy bearing observations are simulated using a random number generator.
3. The noisy measurements are processed by the Kalman filter algorithm to generate estimates of the target's state.

B. TARGET TRACK

As mentioned in Chapter 2, three target track scenarios are investigated: nonmaneuvering, gentle turn, and hard turn. In all three cases the initial target position is (410 nmi, 430 nmi) with X and Y velocities of -400 knots and -380 knots respectively.

C. GENERAL SIMULATION SCENARIO

The overall purpose of this simulation is to be able to track an inbound enemy aircraft before it flies within 300 nmi of the high value unit. By using two sensors which are essentially able to "peek" over the horizon, a long range OTH fighter intercept of the target aircraft may be accomplished. For all simulation runs, an initial target range of 600 nmi is chosen. This allows the sensors to passively track for thirty minutes, and it enables fighter intercept to occur beyond 450 nmi of the HVU, depending on the fighter's initial position and fuel state. The target aircraft is also assumed to be flying inbound at approximately 600 knots.

1. Algorithm Flow

The algorithm can be broken down into the following steps:

1. Define the true target track.
2. Define the observer tracks.
3. Enter a priori estimates $x_{0|-1}$, $P_{0|-1}$, and R_0. Enter bearing error variances σ_1^2 and σ_2^2.
4. Compute the noise free bearings from each sensor to the target for each time interval.
5. Compute random sensor bearing errors using computer generated normal distribution.
6. Add the random bearing errors to the noise free bearings to create noisy bearing measurements.
7. Compute the noisy (x,y) position that results from the intersection of two noisy LOB's.
8. Input this noisy (x,y) measurement into the Kalman filter algorithm.

D. TARGET TRACK

As mentioned in Chapter 2, three target track scenarios are investigated: nonmaneuvering, gentle turn, and hard turn. For all scenarios, the initial target position is (410 nmi, 430 nmi) with x and y velocities of -400 knots and -380 knots respectively. For the scenarios where the target maneuvers, a value is input for the time that the maneuver is to take place. Also, values for the x and y velocities are input for the second leg of the target track. It should be noted that the turn radius of the target is not taken into account in the target track, for at the extreme distances being investigated, target maneuvers almost appear as point turns.

E. OBSERVER TRACKS

The observer tracks each begin at 295 nmi on their respective axis and travel inbound at 420 knots. These observer tracks were chosen to be inbound to represent a more realistic, worst case type scenario. Ideally, it is desired to have both observer tracks going outbound in order to achieve almost perpendicular LOB's. Also, the fuel constraints and maximum on-station time for the observer aircraft are important practical considerations that must be taken into account.

F. INITIALIZATION

In the first series of simulations, different combinations of the initial state estimate $x_{0|-1}$ and the initial error covariance matrix $P_{0|-1}$ are tested. For the first

simulation, and the one by which the other simulations are compared, the a priori state estimate is

$$\hat{\underline{x}}_{o|-1} = \begin{bmatrix} 400 \text{ nmi} \\ 420 \text{ knots} \\ 400 \text{ nmi} \\ 420 \text{ knots} \end{bmatrix}$$

and its associated initial error covariance matrix is

$$P_{o|-1} = \begin{bmatrix} (50\text{nmi})^2 & 0 & 0 & 0 \\ 0 & (50\text{kts})^2 & 0 & 0 \\ 0 & 0 & (50\text{nmi})^2 & 0 \\ 0 & 0 & 0 & (50\text{kts})^2 \end{bmatrix}$$

The initial measurement error covariance matrix for the one degree bearing error case is

$$R_o = \begin{bmatrix} (7\text{nmi})^2 & (5\text{nmi})^2 \\ (5\text{nmi})^2 & (7\text{nmi})^2 \end{bmatrix}$$

G. NOISY BEARING GENERATION

The Box-Müller method is used to generate normally distributed bearing errors. Basically, it is a mapping technique that uses an algebraic identity to establish a one to one relationship between a uniform random variable and a normal random variable. Two random U(0,1) numbers, U_1 and U_2, are transformed into independent N(0,1) random numbers, N_1 and N_2 using the equations

$$N_1 = (-2\ln U_1)^{1/2} \cos 2\pi U_2$$

$$N_2 = (-2\ln U_1)^{1/2} \sin 2\pi U_2$$

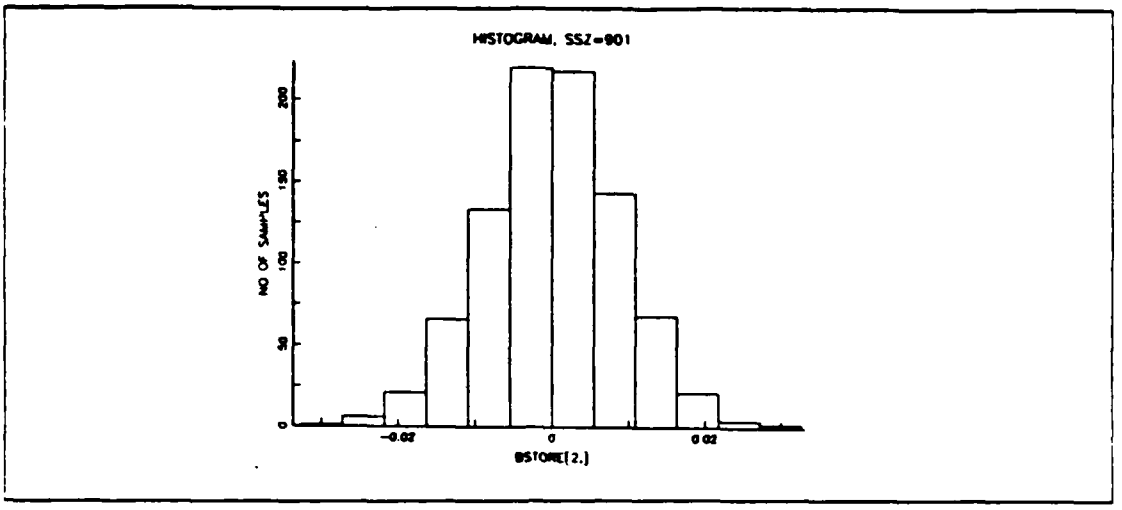

Figure 4.1 Normally Distributed Bearing Error.

Figure 4.1 presents a histogram showing the normal distribution of the bearing error. These normally distributed random numbers are then multiplied by the standard deviation of measurement error for each sensor to produce two independent normally distributed bearing errors for θ_1 and θ_2.

V. SIMULATION RESULTS

A. INTRODUCTION

The purpose of this chapter is to show through various scenarios the effect of different initial state estimates and measurement noise levels on the stability, accuracy, and convergence time of the algorithm. In the following pages, fourteen simulations involving three scenarios are presented. As shown in Figure 2.4, the three scenarios include a nonmaneuvering target track, a target track with a gentle turn, and a track with a hard turn. The first scenario provides the reference with which other cases may be compared. Unless otherwise noted, all simulations use a two second time interval between measurements. Also, all of the simulation results depict the cases of one degree and five degree sensor bearing errors. It should be pointed out that in order to isolate the effect of various parameter changes, a single random number seed is used throughout to represent a specific noise history. There has been no attempt to create statistics based on ensemble of noise histories due to the extreme computational time required.

B. TYPES OF GRAPHS

Five graphs are used in all simulations. These include the x and y positional errors, the x and y velocity errors, and the percent range error. For the case of positional errors, the updated state estimate of x and y position is subtracted from the target's true x and y position. Likewise, for the case of velocity errors, the updated state estimate of velocity is subtracted from the target's true x and y velocity. Range percent error is computed as

$$\text{Range Percent Error} = \frac{|R_T - \hat{R}_T|}{R_T} \times 100$$

where R_T denotes the target's true range from the origin and $\hat{R}_T$ is the updated estimate of target range using the updated state estimate for the x and y positions. In some simulations, the measurement residual error is plotted. The measurement residual is defined as the difference between the actual noisy measurement and the predicted state estimate. It may be expressed as the quantity

$$z_k - H\hat{x}_{k|k-1}$$

C. SCENARIO 1

Scenario 1 consists of ten simulations that demonstrate the effect of various a priori estimates, measurement noise levels, and time intervals between measurements on filter performance. In all ten simulations, the target is on constant course and speed. Figure 5.1 illustrates the target's true track along with noisy measurements. Note by the scale that only a portion of the first quadrant is depicted. In the first simulation, the initial difference between the true target state and the initial state estimate is

$$\underline{x}_{T_0} - \hat{\underline{x}}_{0|-1} = \begin{bmatrix} 10 \text{ nmi} \\ 20 \text{ kt} \\ 30 \text{ nmi} \\ 40 \text{ kt} \end{bmatrix}$$

and the a priori error covariance matrix is

$$P_{0|-1} = \begin{bmatrix} (50_{nmi})^2 & 0 & 0 & 0 \\ 0 & (50kt)^2 & 0 & 0 \\ 0 & 0 & (50_{nmi})^2 & 0 \\ 0 & 0 & 0 & (50kt)^2 \end{bmatrix}$$

Overall, it can be seen that for the one degree bearing error case, the algorithm tracks quite well. Referring to Figure 5.1, the x velocity error initially gets worse before it gets better. It is not until after five minutes have elapsed that the x velocity error is less than the a priori estimate. As can be seen from Figure 5.1, the tracking accuracy for the five degree bearing error case is fair. The one degree bearing error case definitely demonstrates quick and accurate filter convergence; the five degree bearing case seems to meander almost randomly. The sudden rise for the five degree case seems to be an anomalous disturbance. The x and y velocity gains are directly related to bearing accuracy; the higher the accuracy, the greater the gain. For the five degree bearing error case, the velocity gain never exceeds 0.2.

The following nine cases are basically variations of the first case. Table 1 lists the parameters that have been changed for each case.

TABLE 1

SCENARIO PARAMETER CHANGES

FIGURE	TARGET TRACK	TIME INTERVAL	$\hat{\underline{X}}^T_{n-1}$	COMMENTS
5.1 + 5.2	STRAIGHT	2 sec	400 -420 400 -420	REFERENCE CASE $\underline{X}_T = \begin{bmatrix} 410 \\ -400 \\ 430 \\ -380 \end{bmatrix}$ $P_{n-1} = 2500I$
5.3	STRAIGHT	5 sec	400 -420 400 -420	INCREASED TIME INTERVAL
5.4	STRAIGHT	2 sec	430 -540 120 -150	$\underline{X}_T = \begin{bmatrix} 450 \\ -590 \\ 100 \\ -110 \end{bmatrix}$ R matrix correlated
5.5	STRAIGHT	2 sec	430 -540 120 -150	same as above; R matrix uncorrelated
5.6	STRAIGHT	2 sec	400 -150 400 -500	Poor a priori velocity, good position
5.7	STRAIGHT	2 sec	200 -420 480 -420	Poor a priori position, good velocity
5.8	STRAIGHT	2 sec	200 -150 480 -500	Poor a priori position and velocity
5.9	STRAIGHT	2 sec	200 -150 480 -500	same as above; $P_{n-1} = 2500I$
5.10	STRAIGHT	2 sec	400 -410 420 -370	very good a priori est.; $P_{n-1} = 100I$
5.11	STRAIGHT	2 sec	400 -420 400 -420	same as REF. case except $\sigma_r = 5'$, $\sigma_\theta = 1°$
5.12	GENTLE TURN	2 sec	400 -420 400 -420	Tgt turn at 15min; new $\begin{matrix} \dot{x} \\ \dot{y} \end{matrix} = \begin{matrix} -100 \text{ KT} \\ -550 \text{ KT} \end{matrix}$
5.13	HARD TURN	2 sec	400 -420 400 -420	Tgt turn at 15min; new $\begin{matrix} \dot{x} \\ \dot{y} \end{matrix} = \begin{matrix} +310 \text{ KT} \\ -380 \text{ KT} \end{matrix}$
5.14	GENTLE TURN	2 sec	400 -420 400 -420	same as in fig 5.12, 45 minute tracking

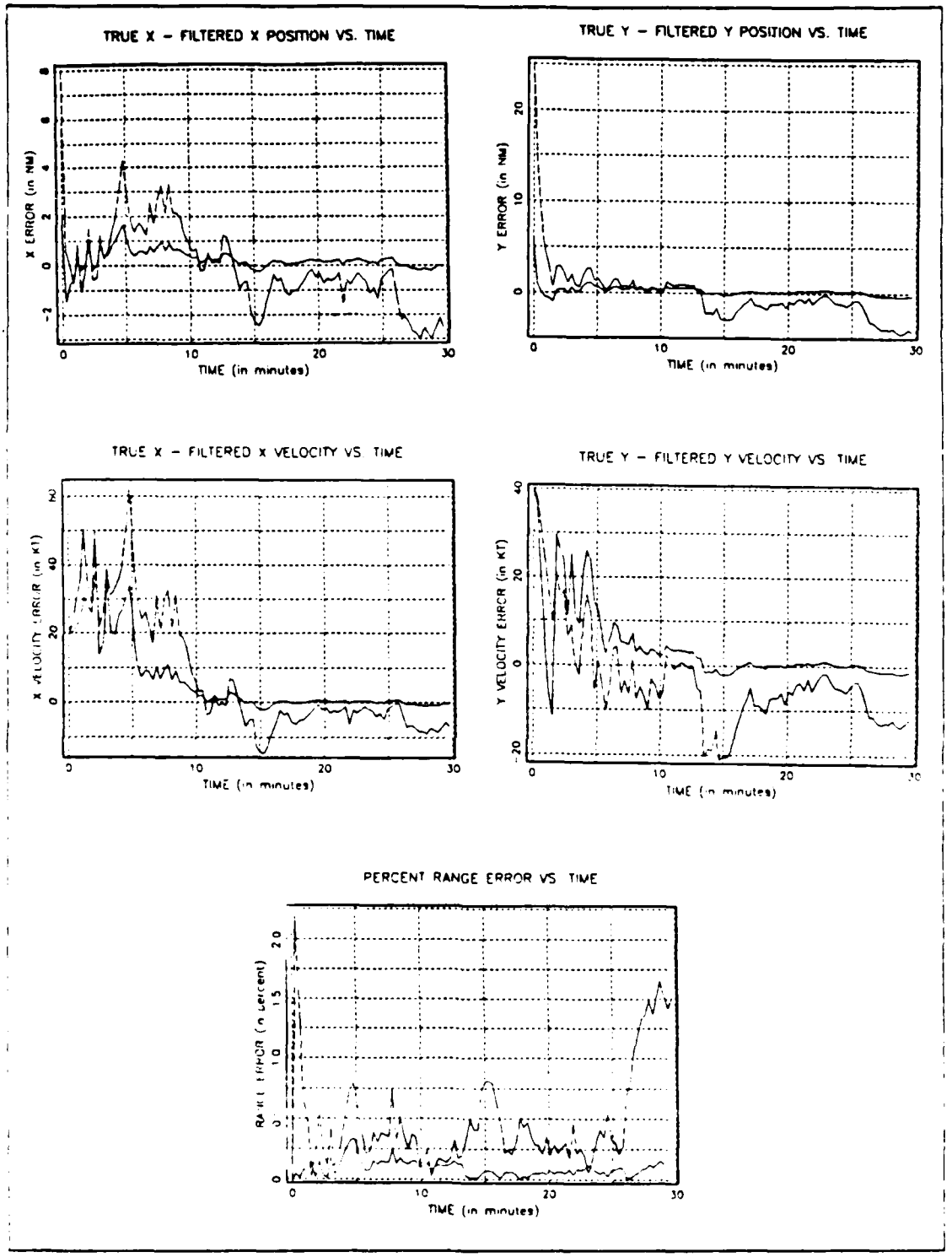

Figure 5.1 Straight Track Reference Case.

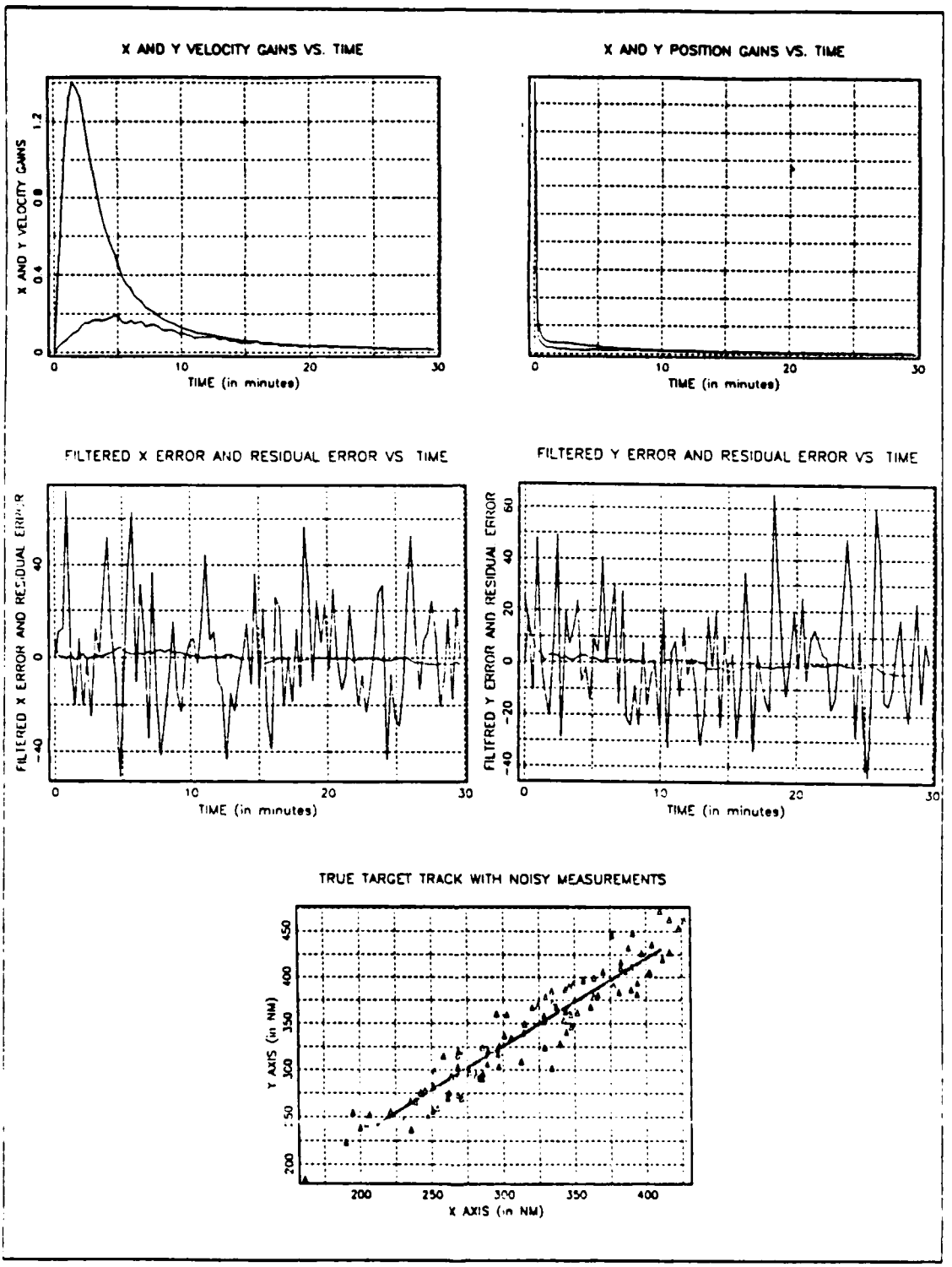

Figure 5.2 Straight Track Reference Case (Cont'd).

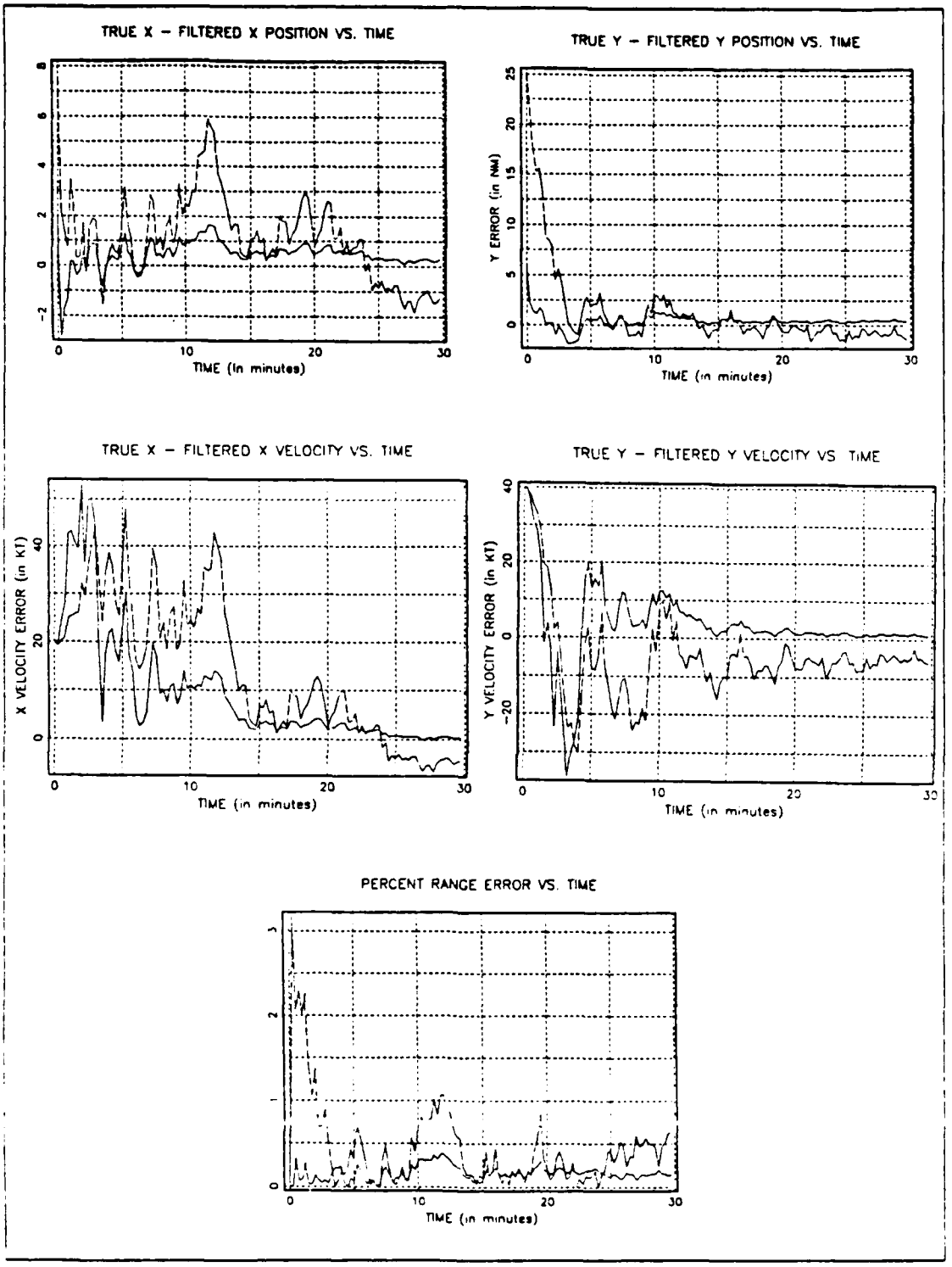

Figure 5.3 Time Interval = 5 sec.

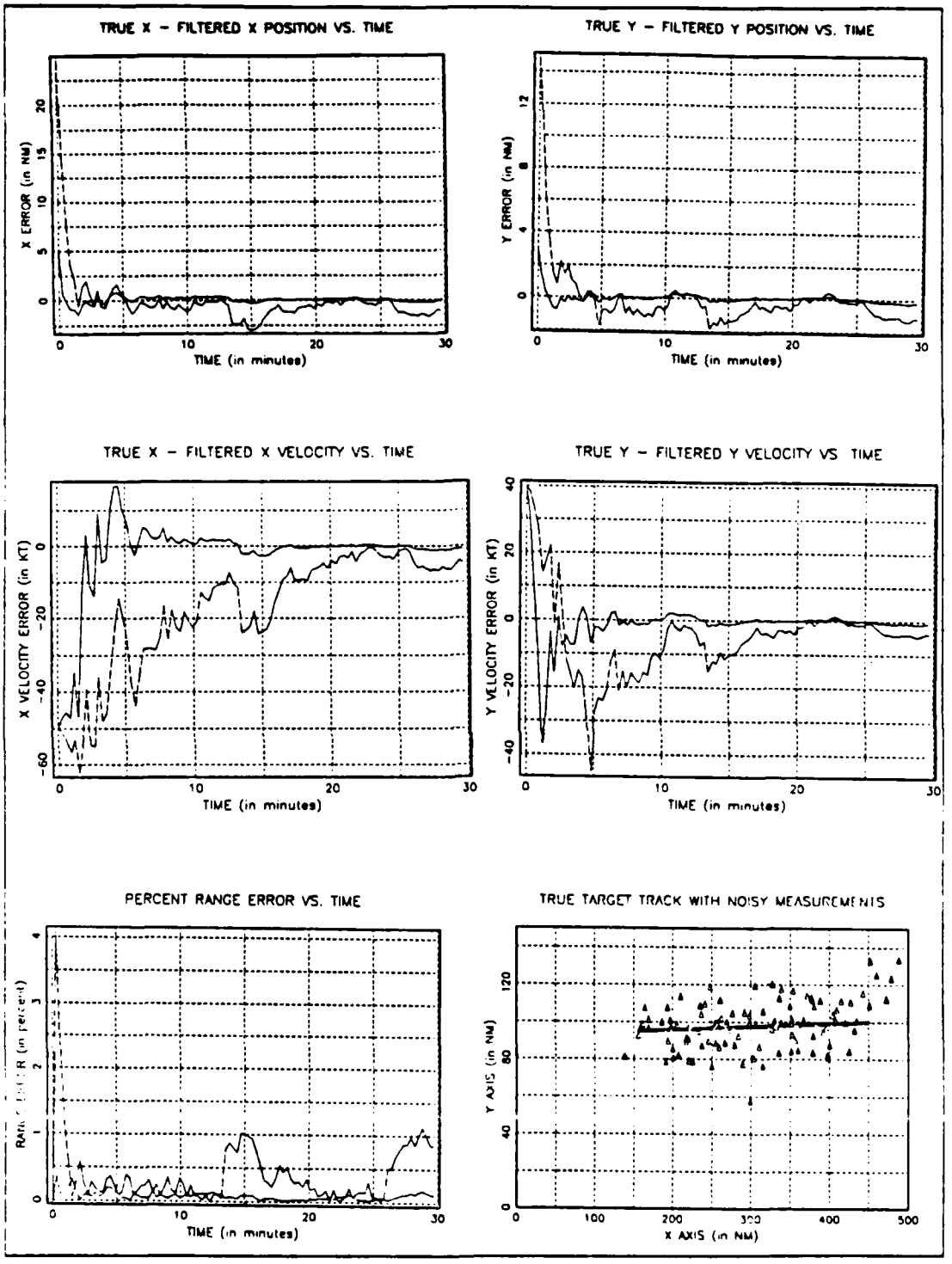

Figure 5.4 APXHAT = 430 -540 120 -50, with R Matrix.

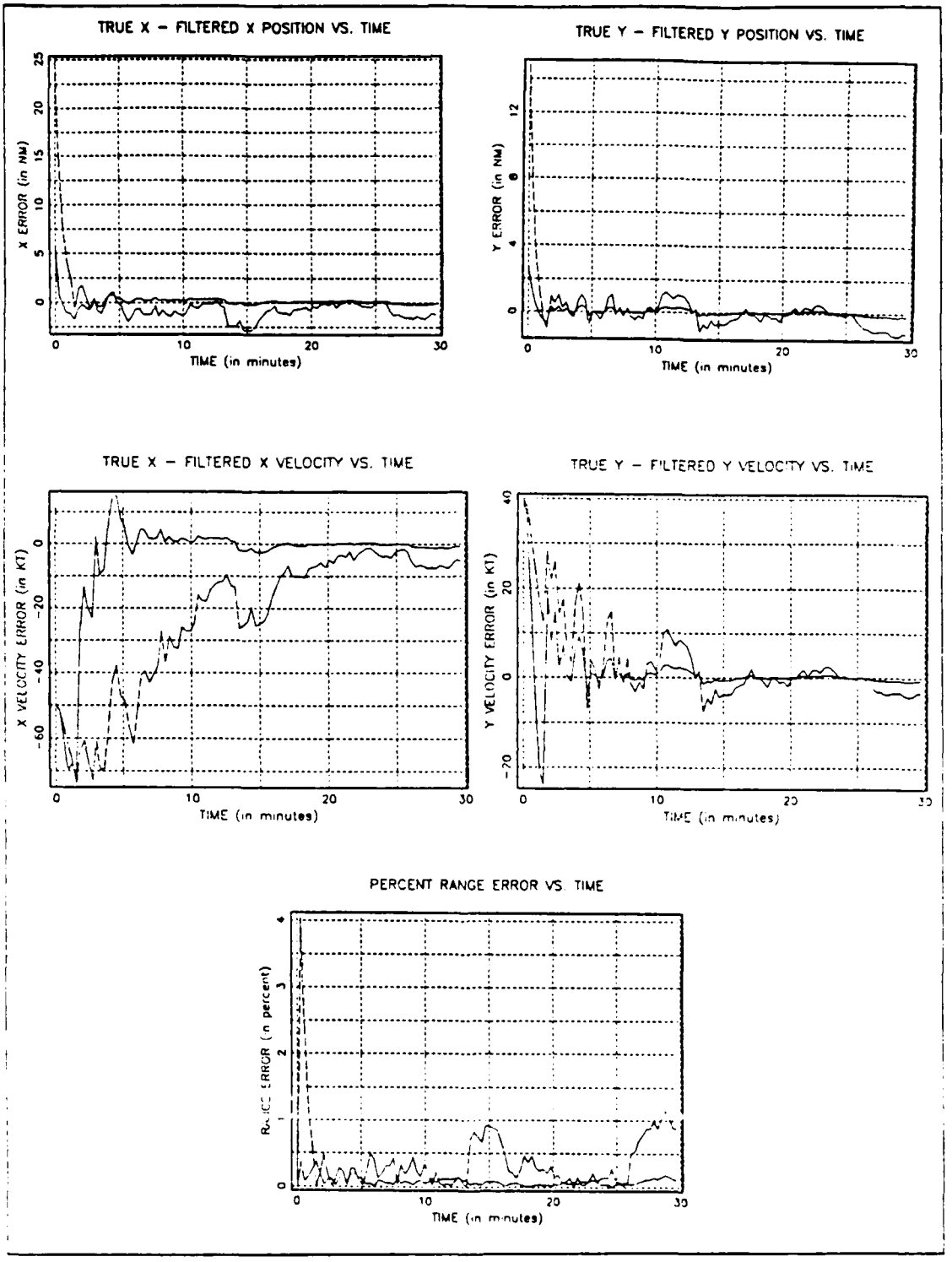

Figure 5.5 Same as Case 3, **Unc**orrelated R Matrix.

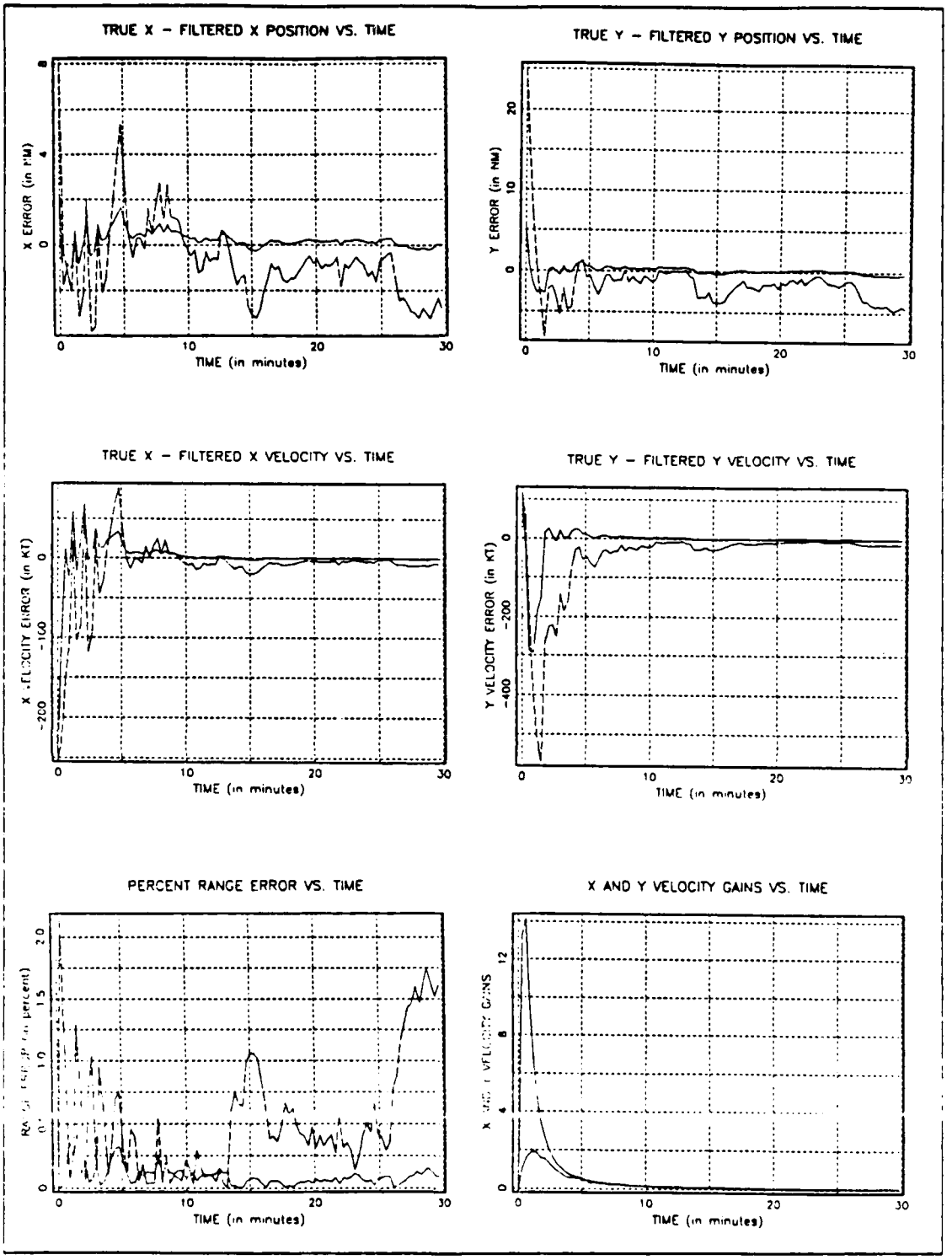

Figure 5.6 APXHAT = 400 -150 400 -500.

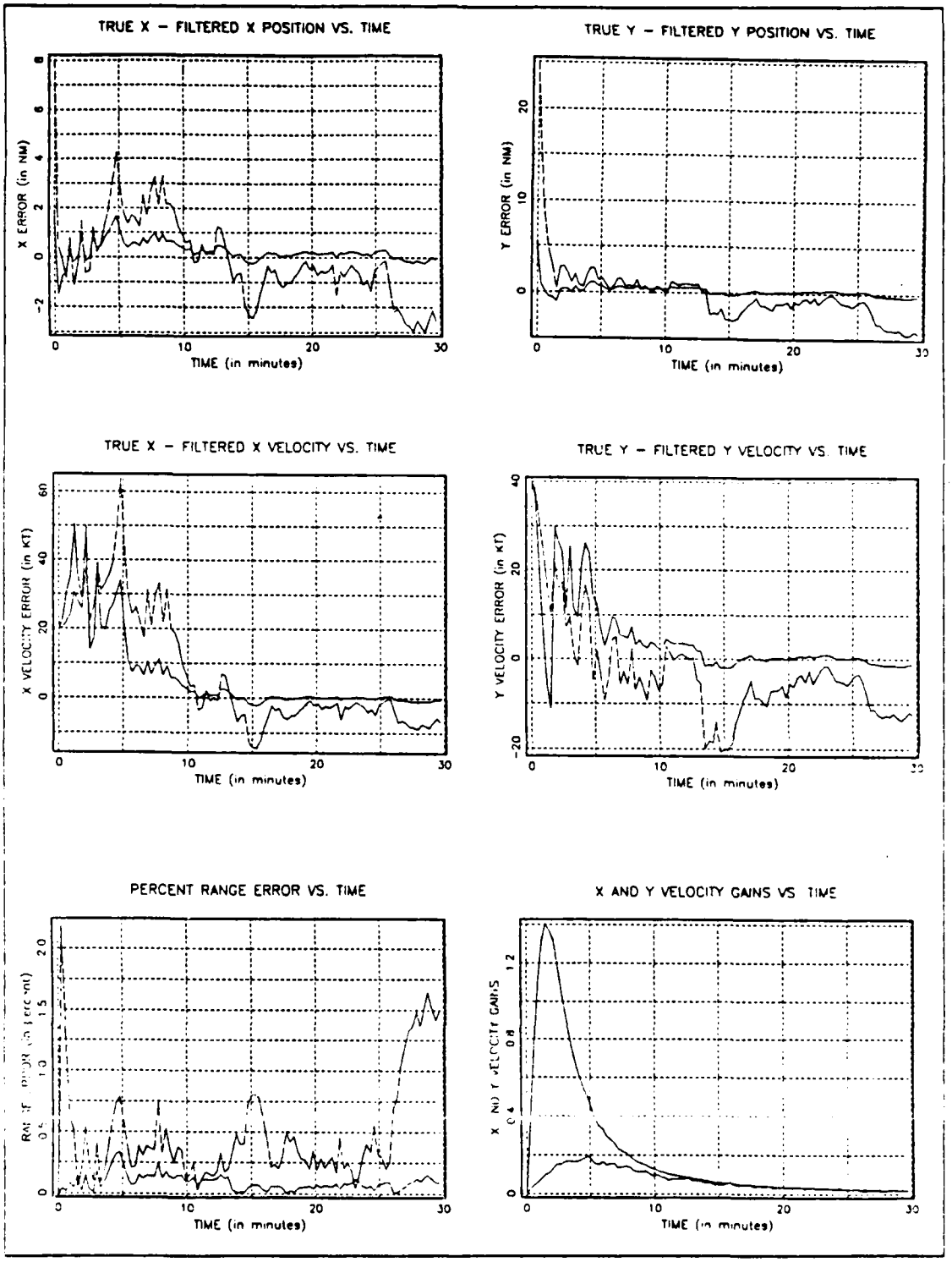

Figure 5.7 APXHAT = 200 -420 480 -420.

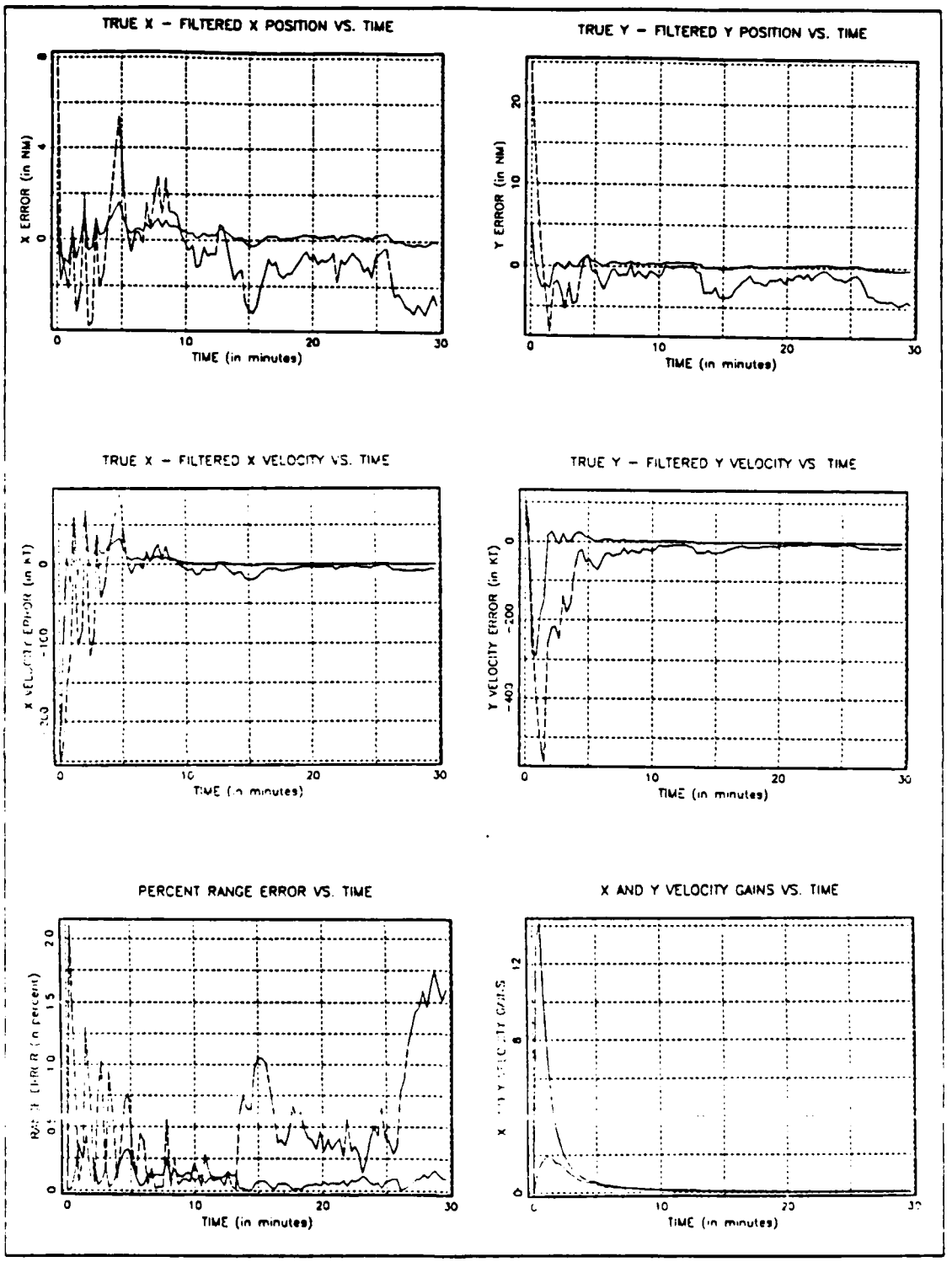

Figure 5.8 APXHAT = 200 -150 480 -500 $P = 10^5 I$.

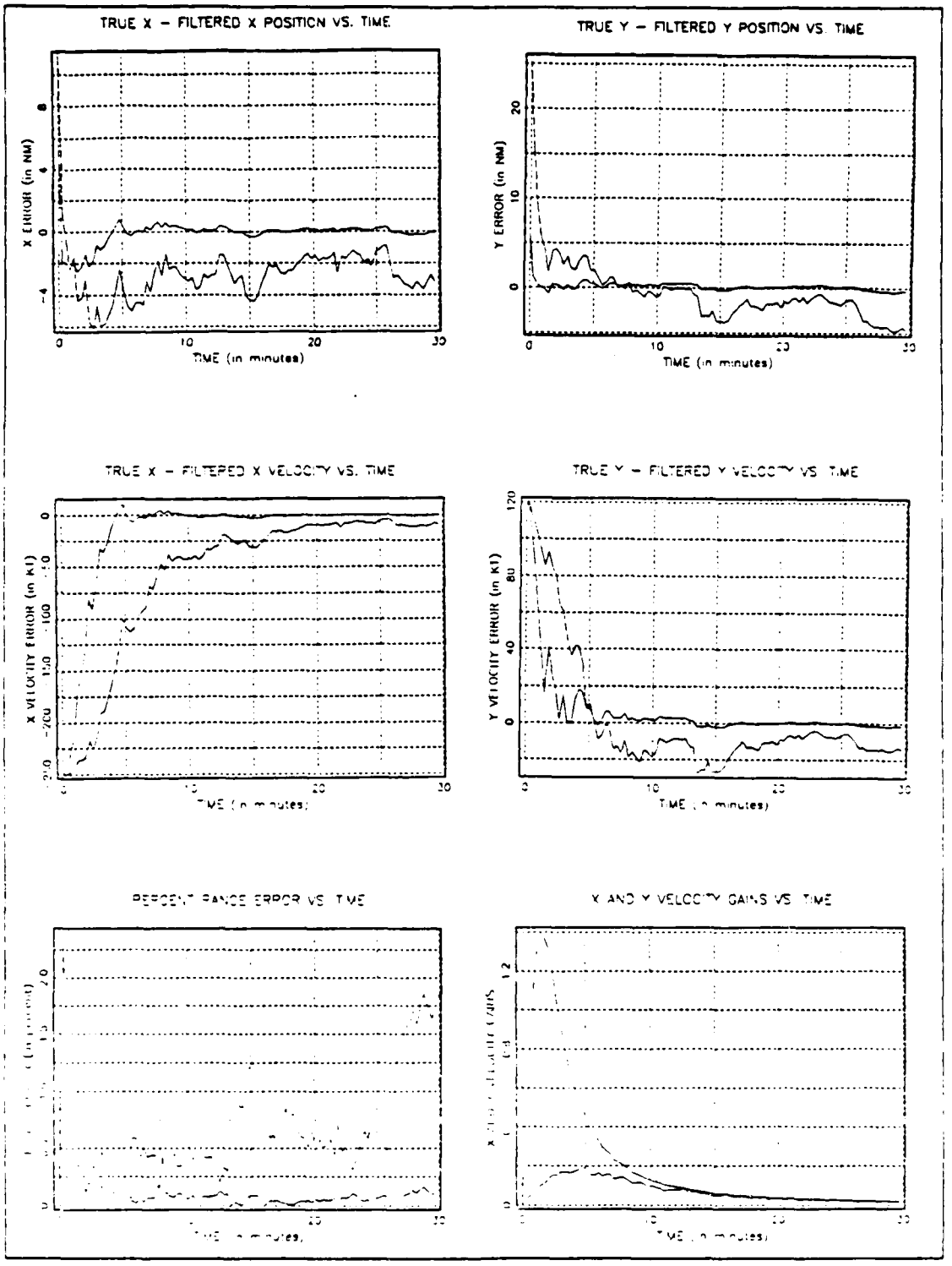

Figure 5.9 APXHAT = 200 -150 480 -500 P = 2500I.

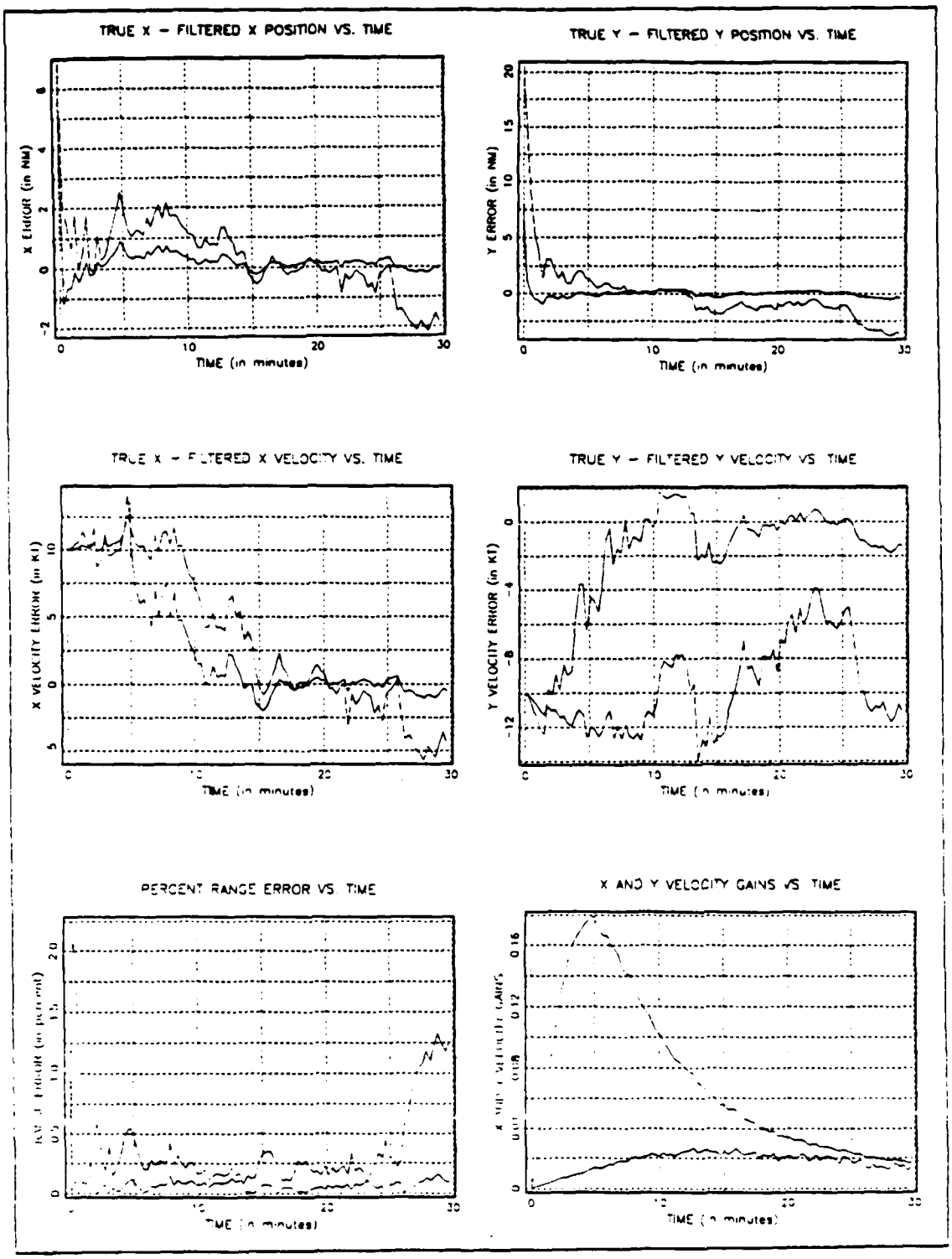

Figure 5.10 APXHAT = 400 -410 420 -370 P = 1001.

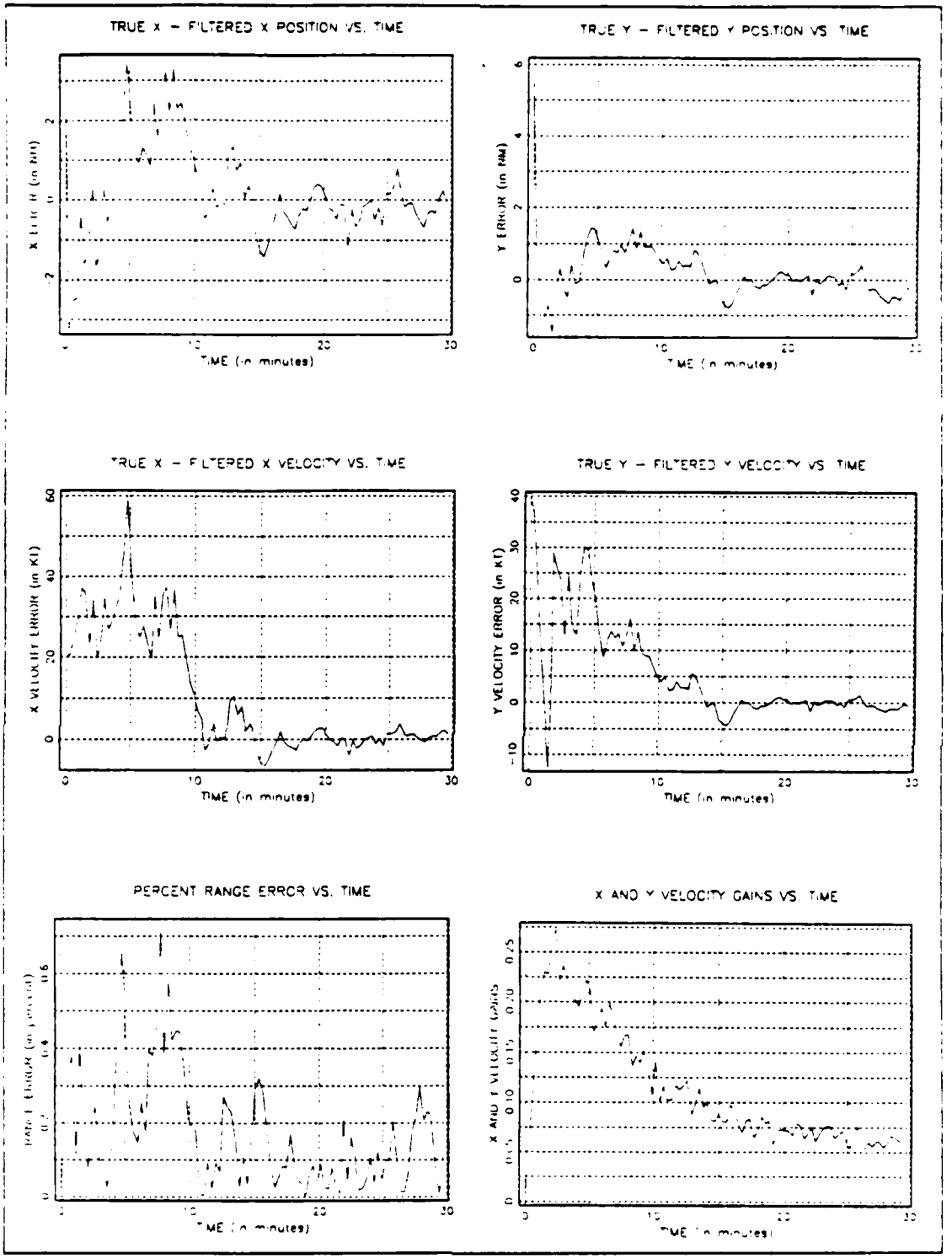

Figure 5.11 σ_1 = 5 deg σ_2 = 1 deg.

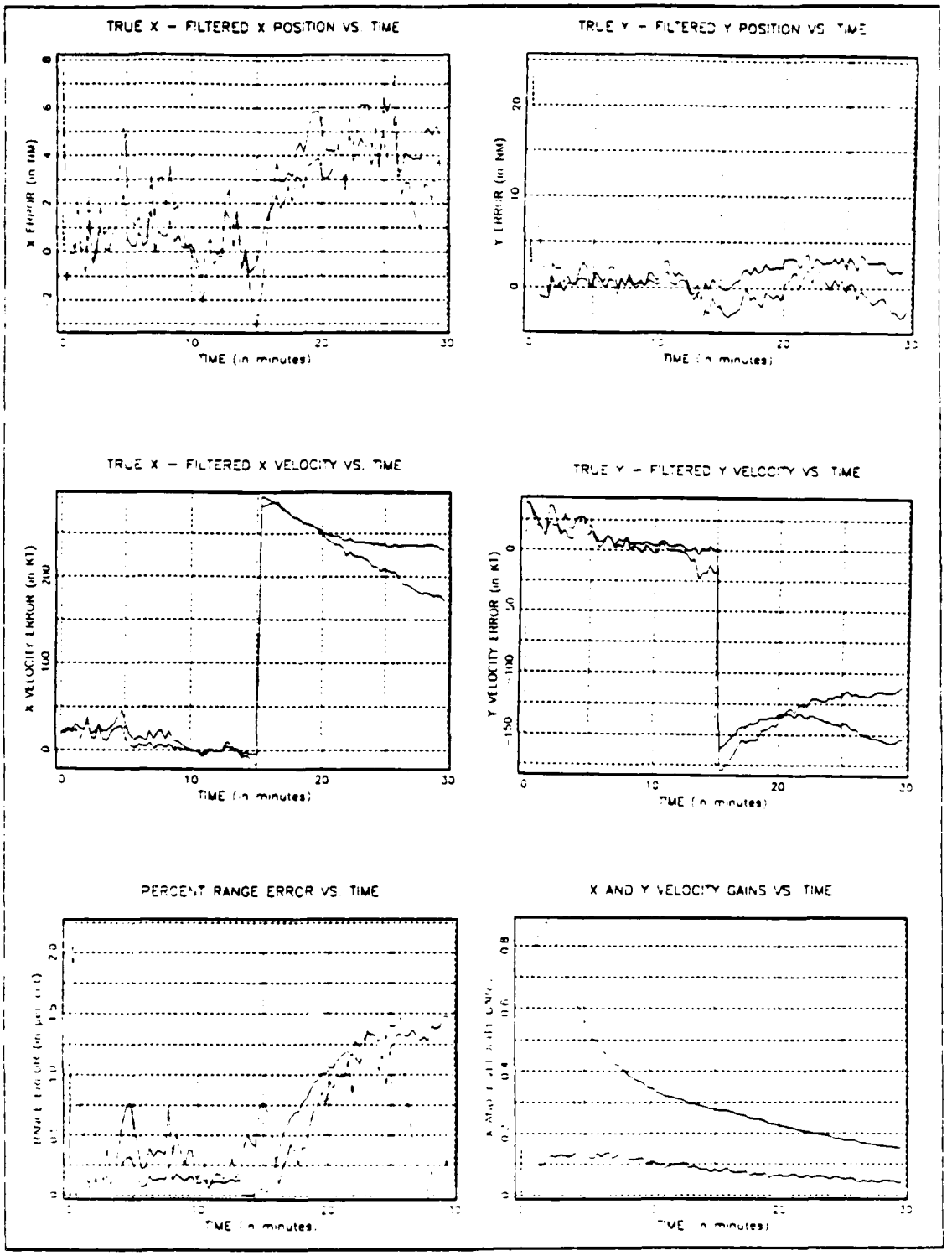

Figure 5.12 Gentle Turn With Q Matrix.

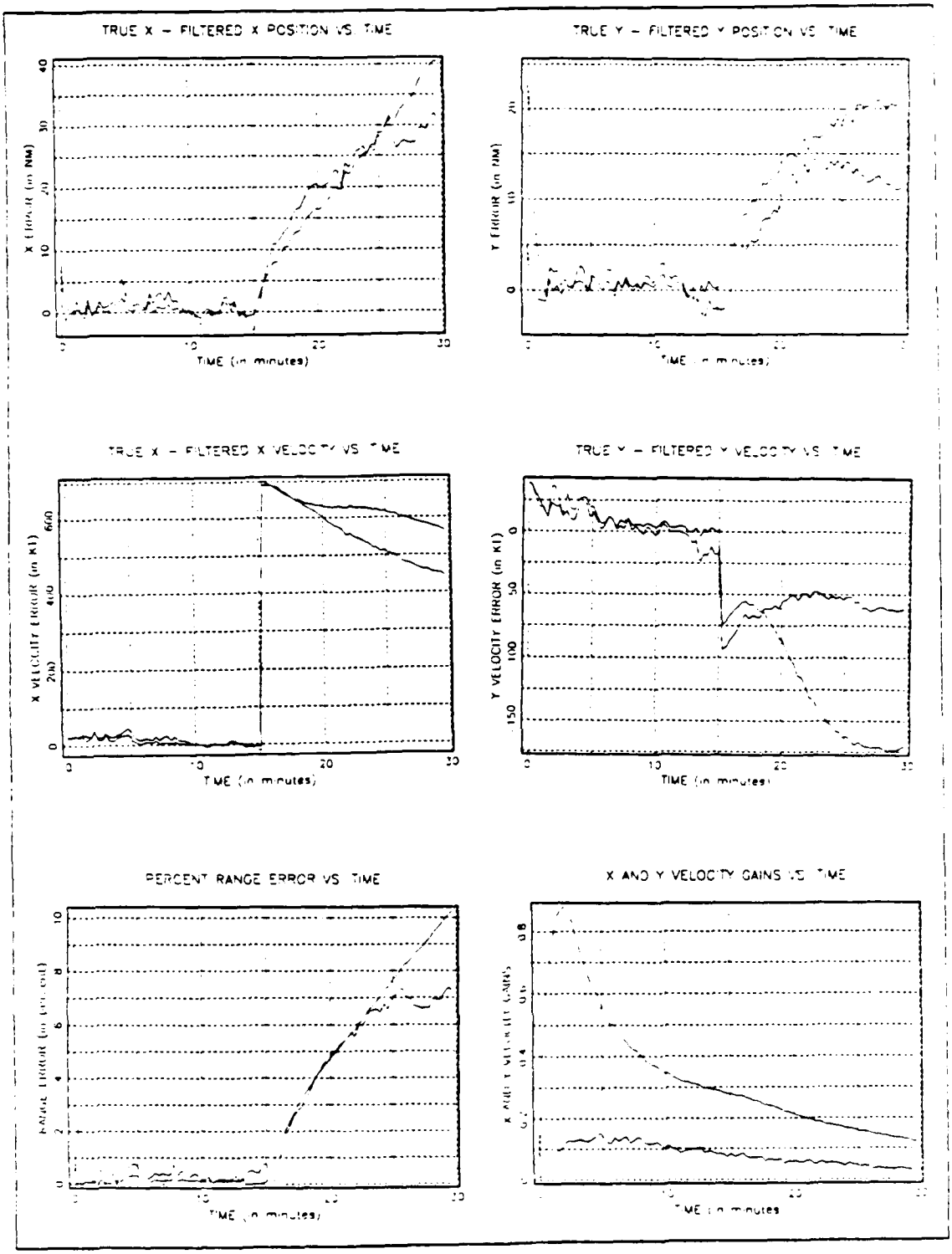

Figure 5.13 Hard Turn With Q Matrix.

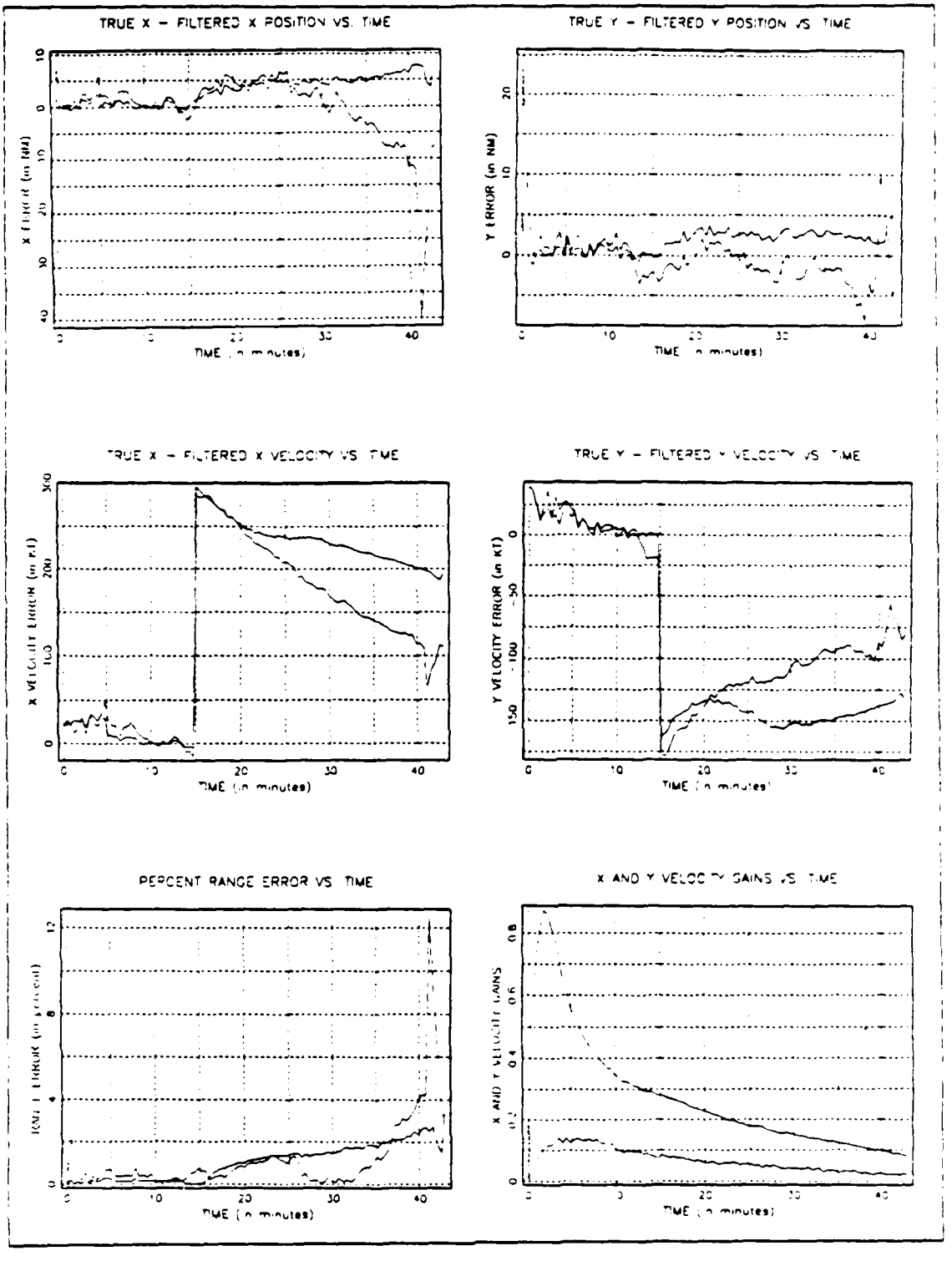

Figure 5.14 Case 11 For 45 min. run.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The purpose of this thesis was to investigate the two sensor bearings only passive tracking problem using Kalman filtering techniques. A computer simulation was developed to generate the target track and the noisy bearing observations from each sensor. Filter performance was exceptional for the nonmaneuvering target case. With one degree sensor bearing error and two second measurement intervals, the filter was able to consistently track the target to within one quarter of one percent range error in the first five minutes. As was expected, filter accuracy was degraded as bearing error was increased. The tracker performed reasonably well for sensor bearing errors as high as eight degrees.

For the case of a maneuvering target, filter performance was marginal. Filter convergence to an accuracy attained prior to the target maneuver did not occur. The use of a state excitation covariance matrix by itself was not sufficient enough to properly account for target maneuvers. What is needed is a reliable zig detector that quickly recognizes target maneuvers so that the filter gains may be reinitialized. The problem of detecting target maneuvers is not trivial. At long ranges, error ellipses may be twenty to forty times larger than the actual distance the target has moved. Sifting out a bona fide target maneuver from these extremely noisy measurements is quite difficult. Determining a target maneuver by attempting to find a pattern in the measurement residuals is only successful if the time between measurements is greater than thirty seconds. A drawback to this approach is that filter accuracy is degraded because fewer measurements are being processed.

From the simulation runs it was found that the most influential factors in determining tracking accuracy and speed of convergence were the sensor bearing accuracies and the time between measurements. Factors that contributed to a lesser extent included the accuracy of the initial state estimate along with its associated degree of confidence and the positions of the sensors. Inaccurate a priori information did not degrade the filter's accuracy, it only increased the time for convergence. Filter accuracy improved as the lines of bearing came closer to being perpendicular.

B. RECOMMENDATIONS

This study is by no means complete. Some areas for further study include the following:

1. Run an entire ensemble of simulations (perhaps 1000 runs) to generate reliable statistics on the filter's performance.
2. Investigate more fully a method to detect target maneuvers so that adaptive control techniques may be used to alleviate the problem of filter divergence.
3. Examine the utility of using more sensors to cover a comparable sector. Are more sensors necessarily beneficial?
4. Perform the simulation using a different coordinate system (such as polar coordinates) and compare the results to those obtained using a Cartesian model.

APPENDIX A
DERIVATION OF MEASUREMENT ERROR COVARIANCE MATRIX

In this appendix the measurement error covariance matrix $\mathbf{R}_k$ is derived. Recalling the measurement equation from Chapter 3, the measurement noise vector v_k is assumed to be Gaussian and zero mean with variance $\mathbf{R}_k$. That is,

$$v_k \sim N(0,\mathbf{R}_k) \qquad (A\text{-}1)$$

The purpose of $\mathbf{R}_k$ is to statistically describe the noisiness of the x and y measurements obtained through the intersection of noisy lines of bearing. Basically, the 2 by 2 measurement error covariance matrix describes this noisiness by displaying the variance and covariance of the noisy x and y measurements in terms of each sensor's bearing measurement and accuracy. Referring to Figure A.1, the position (X_T,Y_T) represents a possible true position of the target based on noisy sensor bearing observations, θ_1 and θ_2. The position of the target (X_T, Y_T) is a jointly distributed random variable whose expected value coincides with the intersection of the bearing lines.

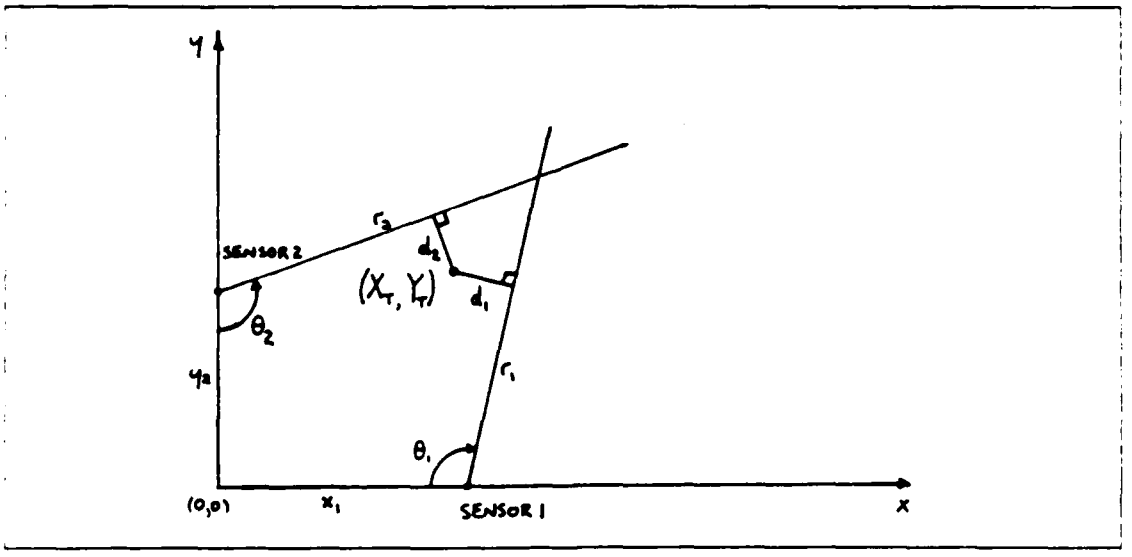

Figure A.1 Target Observer Geometry Revisited.

To develop the relationship between X_T and Y_T, the LOB's from each sensor may be expressed in the general form for the equation of a line:

$$x \sin\theta_1 + y \cos\theta_1 - x_1 \sin\theta_1 = 0 \qquad (A\text{-}2)$$

$$x \cos\theta_2 + y \sin\theta_2 - y_2 \sin\theta_2 = 0 \qquad (A\text{-}3)$$

The distance from the position (X_T, Y_T) to each sensor line of bearing is denoted by d_1 and d_2 for each sensor LOB respectively. From the problem geometry and by using the equation for the distance between a point and a line, the displacement distances d_1 and d_2 may be expressed as

$$d_1 = \pm\left[(X_T - x_1)\sin\theta_1 + Y_T \cos\theta_1\right] \qquad (A\text{-}4)$$

$$d_2 = \pm\left[X_T \cos\theta_2 + (Y_T - y_2)\sin\theta_2\right] \qquad (A\text{-}5)$$

Since both sensor bearing errors are assumed to be Gaussian zero mean random variables with bearing variances σ_1^2 and σ_2^2, it follows that the displacement random variables d_1 and d_2 are also zero mean Gaussian with displacement variances σ_1^2 and σ_2^2 respectively. Figure A.2 illustrates the relationship between the displacement variance and the sensor bearing error variance.

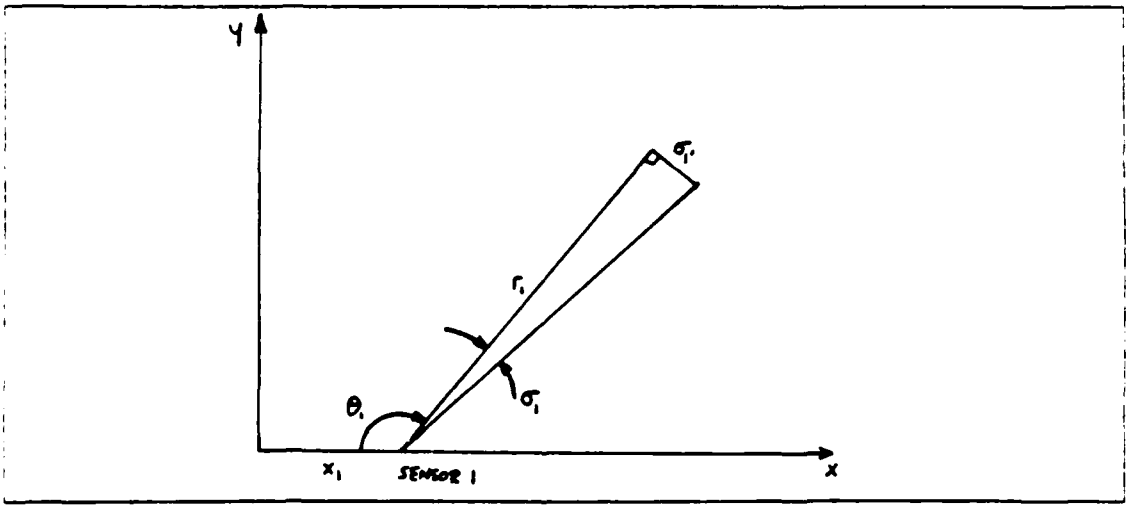

Figure A.2 Relationship Between Bearing and Displacement Errors.

In this figure, the bearing error standard deviation for sensor 1, σ_1, is expressed in degrees or radians whereas the displacement error standard deviation for sensor 1, σ_1 , is expressed in nautical miles. They are related by

$$\sigma_{1'} = r_1 \tan \sigma_1 \qquad (A\text{-}6)$$

where r_1 is the approximate distance from sensor 1 to the target in nautical miles.

Therefore, the displacement distances d_1 and d_2 are normal random variables that may be written as

$$d_1 \sim N(0, \sigma_{1'}^2) \qquad (A\text{-}7)$$

$$d_2 \sim N(0, \sigma_{2'}^2) \qquad (A\text{-}8)$$

Having described the displacement random variables d_1 and d_2 from sensor 1 and sensor 2 lines of bearing, equations A-4 and A-5 may be rewritten as

$$X_T \sin\theta_1 + Y_T \cos\theta_1 = x_1 \sin\theta_1 + N_1 \qquad (A\text{-}9)$$

$$X_T \cos\theta_2 + Y_T \sin\theta_2 = y_2 \sin\theta_2 + N_2 \qquad (A\text{-}10)$$

where $N_1 = N(0, \sigma_{1'}^2)$ and $N_2 = N(0, \sigma_{2'}^2)$. Solving equations (A-9) and (A-10) for X_T and Y_T yields:

$$X_T = \frac{y_2 \sin\theta_2 \cos\theta_1 - x_1 \sin\theta_1 \sin\theta_2 + N_2 \cos\theta_1 - N_1 \sin\theta_2}{\cos(\theta_1 + \theta_2)}$$

$$Y_T = \frac{x_1 \sin\theta_1 \cos\theta_2 - y_2 \sin\theta_2 \sin\theta_1 + N_1 \cos\theta_2 - N_2 \sin\theta_1}{\cos(\theta_1 + \theta_2)}$$

or $$X_T = A_1 + A_2 N_1 + A_3 N_2 \qquad (A\text{-}11)$$

$$Y_T = B_1 + B_2 N_1 + B_3 N_2 \qquad (A\text{-}12)$$

where $$A_1 = \frac{y_2 \sin\theta_2 \cos\theta_1 - x_1 \sin\theta_1 \sin\theta_2}{\cos(\theta_1 + \theta_2)}$$

and $$A_2 = \frac{-\sin\theta_2}{\cos(\theta_1+\theta_2)} \qquad A_3 = \frac{\cos\theta_1}{\cos(\theta_1+\theta_2)}$$

$$B_1 = \frac{x_1 \sin\theta_1 \cos\theta_2 - y_1 \sin\theta_1 \sin\theta_2}{\cos(\theta_1+\theta_2)}$$

$$B_2 = \frac{\cos\theta_2}{\cos(\theta_1+\theta_2)} \qquad B_3 = \frac{-\sin\theta_1}{\cos(\theta_1+\theta_2)}$$

Note that since X_T and Y_T are linear combinations of normally distributed random variables, they must also be normally distributed. That is,

$$X_T \sim N(\mu_{X_T}, \sigma_{X_T}^2) \qquad (A\text{-}13)$$

$$Y_T \sim N(\mu_{Y_T}, \sigma_{Y_T}^2) \qquad (A\text{-}14)$$

where $\mu_{X_1} = A_1$

$$\sigma_{X_T}^2 = A_2^2 \sigma_{1'}^2 + A_3^2 \sigma_{2'}^2 \qquad (A\text{-}15)$$

and $\mu_{Y_1} = B_1$

$$\sigma_{Y_T}^2 = B_2^2 \sigma_{1'}^2 + B_3^2 \sigma_{2'}^2 \qquad (A\text{-}16)$$

Now, the measurement error covariance matrix **R** may be written as

$$R = \begin{bmatrix} \mathrm{Var}(X_T) & \mathrm{Cov}(X_T, Y_T) \\ \mathrm{Cov}(X_T, Y_T) & \mathrm{Var}(Y_T) \end{bmatrix} \qquad (A\text{-}17)$$

By definition, the covariance of the random variables X_T and Y_T is the expected value of their product minus the product of their means. That is,

$$\mathrm{Cov}(X_T, Y_T) = E[X_T Y_T] - \mu_{X_T}\mu_{Y_T} \qquad (A\text{-}18)$$

Substituting equations (A-11) and (A-12) into the above equation yields

$$\mathrm{Cov}(X_T, Y_T) = E[(A_1 + A_2 N_1 + A_3 N_2)(B_1 + B_2 N_1 + B_3 N_2)] - A_1 B_1 \qquad (A\text{-}19)$$

By noting that N_1 and N_2 are zero mean, the expected value of their product is the product of their means. That is, $E[N_1 N_2] = E[N_1]\, E[N_2] = 0$. Using this fact, the covariance between the random variables X_T and Y_T is simplified to

$$\mathrm{Cov}(X_T, Y_T) = A_2 B_2 \sigma_{1'}^2 + A_3 B_3 \sigma_{2'}^2 \qquad \text{(A-20)}$$

Substituting equations (A-15), (A-16), and (A-20) into equation (A-17) enables the measurement error covariance matrix **R** to be written as

$$R = \begin{bmatrix} A_2^2 \sigma_{1'}^2 + A_3^2 \sigma_{2'}^2 & A_2 B_2 \sigma_{1'}^2 + A_3 B_3 \sigma_{2'}^2 \\ A_2 B_2 \sigma_{1'}^2 + A_3 B_3 \sigma_{2'}^2 & B_2^2 \sigma_{1'}^2 + B_3^2 \sigma_{2'}^2 \end{bmatrix}$$

Lastly, substituting the values for A_2 and A_3 into the above equation yields the final expression for the measurement error covariance matrix

$$R = \begin{bmatrix} \dfrac{\sigma_{1'}^2 \sin^2\theta_2 + \sigma_{2'}^2 \cos^2\theta_1}{\cos^2(\theta_1+\theta_2)} & \dfrac{-\sigma_{1'}^2 \sin\theta_2\cos\theta_2 - \sigma_{2'}^2 \cos\theta_1 \sin\theta_1}{\cos^2(\theta_1+\theta_2)} \\ \dfrac{-\sigma_{1'}^2 \sin\theta_2\cos\theta_2 - \sigma_{2'}^2 \cos\theta_1 \sin\theta_1}{\cos^2(\theta_1+\theta_2)} & \dfrac{\sigma_{1'}^2 \cos^2\theta_2 + \sigma_{2'}^2 \sin^2\theta_1}{\cos^2(\theta_1+\theta_2)} \end{bmatrix}$$

APPENDIX B
SIMULATION PROGRAM LISTING

```
      ∇ KALMAN
[1]   ⍝ THIS PROGRAM IS A TWO-SENSOR KALMAN FILTER TRACKING ALGORITHM.
[2]   ⍝
[3]   ⍝
[4]   ⍝THE FOLLOWING IS A LIST OF THE PRINCIPAL VARIABLES USED:
[5]   ⍝APXHAT.....A PRIORI STATE ESTIMATE XHAT
[6]   ⍝BE1,BE2.....NORMALLY DISTRIBUTED BEARING ERRORS
[7]   ⍝DT.....TIME STEP (IN HOURS)
[8]   ⍝ERR.....ACTUAL POSITION ERROR (TRUE-PREDICTED)
[9]   ⍝G.....KALMAN GAIN
[10]  ⍝H.....MEASUREMENT MATRIX
[11]  ⍝NTHETA1,NTHETA2..NOISY BEARINGS FROM SENSORS 1 AND 2 TO TARGET
[12]  ⍝NZ.....MEASURED X,Y POSITION (NOISY)
[13]  ⍝P.....ERROR COVARIANCE MATRIX
[14]  ⍝PHI.....STATE TRANSITION MATRIX
[15]  ⍝Q.....STATE EXCITATION COVARIANCE MATRIX
[16]  ⍝R.....MEASUREMENT NOISE COVARIANCE MATRIX
[17]  ⍝RNGPCERR.....PERCENT RANGE ERROR
[18]  ⍝R2D.....RADIANS TO DEGREES
[19]  ⍝TKM.....TIME VECTOR (IN MINUTES)
[20]  ⍝VX,VY.....TRUE TARGET VELOCITIES IN X AND Y DIRECTIONS
[21]  ⍝VX1,VY2.....SENSOR 1 AND SENSOR 2 VELOCITIES
[22]  ⍝XHAT.....STATE ESTIMATE
[23]  ⍝XTK,YTK.....VECTORS OF TRUE TARGET POSITIONS
[24]  ⍝X1K,Y2K.....VECTORS OF TRUE SENSOR POSITIONS
[25]  ⍝
[26]  ⍝*****************************************************************
[27]  ⍝
[28]  ⍝ INITIAL CONDITION INPUTS:
[29]  ⍝
[30]  ⍝'INPUT DESIRED RANDOM LINK:'
[31]   RLSAVE←⎕RL←265067500
[32]  ⍝
[33]   'HOW MANY ITERATIONS ARE TO BE RUN (K)?'
[34]   K←⎕
[35]  ⍝
[36]   'ENTER TRUE TARGET PARAMETERS: XT0,VXF,VXS,YT0,VYF,VYS'
[37]   TRUESTATE←⎕
[38]   XT0←TRUESTATE[1]
[39]   VX←TRUESTATE[2 3]
[40]   YT0←TRUESTATE[4]
[41]   VY←TRUESTATE[5 6]
[42]  ⍝
[43]   'WHEN IS THE TARGET GOING TO TURN (WHICH ITERATION NO.):'
[44]   TURN←⎕
[45]  ⍝
[46]   'NEXT ENTER SENSOR 1 AND SENSOR 2 POSITION AND VELOCITY'
[47]   'DATA AS X10,VX1,Y20,VY2'
[48]   SENSPOS←⎕
[49]   X10←SENSPOS[1]
[50]   VX1←SENSPOS[2]
[51]   Y20←SENSPOS[3]
[52]   VY2←SENSPOS[4]
[53]  ⍝
[54]  ⍝
[55]   'TIME STEP TO BE USED (DT,IN SECONDS)?'
[56]   DT←⎕
[57]   DTS←DT
[58]   DT←DT÷3600
[59]  ⍝
[60]  ⍝'NO. OF POINTS TO INCLUDE IN THE REGRESSION:'
```

```
[61] ⍝NRP←⍬
[62] ⍝
[63]  'NOW ENTER ALL A PRIORI ESTIMATES AND MATRICES:'
[64]  ' '
[65] ⍝
[66]  'INITIAL GUESS XHAT (FOUR ELEMENTS MUST BE ENTERED):'
[67]  XHAT←⍬
[68]  XHAT←4 1⍴XHAT
[69]  APXHAT←XHAT
[70]  ' '
[71] ⍝
[72]  'INITIAL P MATRIX DIAGONAL ELEMENTS (FOUR ENTRIES):'
[73]  DIAGONAL←⍬
[74]  P←((⍳4)∘.=⍳4)×4 4⍴DIAGONAL
[75]  ' '
[76] ⍝
[77] ⍝'ENTER OFF-DIAGONAL POSITIONAL AND VELOCITY COVARIANCES (2 INPUTS):'
[78]  OFFDIAG←0 0
[79]  P[1;3]←P[3;1]←OFFDIAG[1]
[80]  P[2;4]←P[4;2]←OFFDIAG[2]
[81]  PSAVE←P
[82] ⍝
[83]  'INITIAL R MATRIX (4 NUMBERS IN THE ORDER UL,UR,LL,LR):'
[84]  R←⍬
[85]  RSAVE←R←2 2⍴R
[86] ⍝
[87] ⍝SET UP SOME VECTORS NEEDED TO GENERATE THE Q MATRIX.
[88] ⍝C←((○1)÷(4×360⌶))⌶*2
[89] ⍝F1←1 2 1 2 2 4 2 4
[90] ⍝F1←F1,-F1
[91] ⍝F2←4 4 2 2 4 4 ,10⍴2
[92] ⍝F3←(8⍴4),8↑F2
[93] ⍝F4←4 3 4 3 3 2 3 2
[94] ⍝F4←F4,F4
[95] ⍝
[96]  ' '
[97]  'ENTER MAX BEARING ERROR IN DEG. FOR THETA1 AND THETA2 (≥2 INPUTS):'
[98]  B6←⍬
[99] ⍝
[100] ⍝*****************************************************************
[101] ⍝
[102] ⍝ INITIALIZED STORAGE VECTORS FOR GRAPHING PURPOSES ONLY
[103] ⍝
[104]  QGVSTORE←QGSTORE←QVERRSTORE←QERRSTORE←QRESIDERRSTORE←(2,K+1)⍴0
[105]  QRNGPCERRSTORE←QMSAVE←(1,K+1)⍴0
[106] ⍝
[107] ⍝ THE FOLLOWING OUTER LOOP IS FOR GRAPHING THREE SETS OF SENSOR BEARING
[108] ⍝ ERRORS ON THE SAME GRAPH. THE TRACKING ALGORITHM IS, IN EFFECT, BEING
[109] ⍝ RUN THREE TIMES, WITH EACH RUN USING A DIFFERENT PAIR OF SENSOR
[110] ⍝ BEARING ERRORS.
[111] ⍝
[112]  MAINLOOP←1
[113] TOP:BRGERR←B6[(¯1+2×MAINLOOP),2×MAINLOOP]
[114]  P←PSAVE
[115]  R←RSAVE
[116]  XHAT←APXHAT
[117]  A1←BRGERR[1]
[118]  A2←BRGERR[2]
[119] ⍝
[120] ⍝SET UP Q MATRIX INITIALLY TO BE A 4X4 MATRIX OF ZEROS.
[121]  Q←4 4⍴0
[122] ⍝
[123] ⍝*****************************************************************
[124] ⍝
[125] ⍝ THE PURPOSE OF THIS PROGRAM SEGMENT IS TO
[126] ⍝ ESTABLISH TRUE TARGET TRACK (WHICH INCLUDES TWO LEGS) AND SET UP
[127] ⍝ SENSOR 1 AND SENSOR 2 TRACKS. NOISE-FREE BEARINGS FROM EACH SENSOR
[128] ⍝ TO THE TARGET ARE COMPUTED. NOISY ZERO-MEAN NORMALLY DISTRIBUTED
[129] ⍝ BEARING ERRORS ARE GENERATED AND THEN ADDED TO THE NOISE-FREE
[130] ⍝ BEARINGS. FROM THE INTERSECTION OF THESE NOISY LOB'S, THE NOISY
```

```
[131] ⍝ (X,Y) POSITION OF THE TARGET IS COMPUTED AND STORED FOR LATER USE
[132] ⍝ IN THE TRACKING FILTER AS THE TARGET'S 'MEASURED' POSITION FOR
[133] ⍝ THE KTH TIME ITERATION.
[134] ⍝
[135] ⍝ SET UP SOME INITIAL CALCULATIONS:
[136]  TK←DT×0,⍳K
[137]  TKM←TK×60
[138] ⍝
[139]  XTKF←XT0+VX[1]×DT×0,⍳TURN
[140]  XTKS←(¯1↑XTKF)+VX[2]×DT×⍳(K-TURN)
[141]  XTK←XTKF,XTKS
[142]  YTKF←YT0+VY[1]×DT×0,⍳TURN
[143]  YTKS←(¯1↑YTKF)+VY[2]×DT×⍳(K-TURN)
[144]  YTK←YTKF,YTKS
[145] ⍝
[146]  X1K←X10+VX1×TK
[147]  Y2K←Y20+VY2×TK
[148] ⍝
[149]  R2D←180÷○1
[150]  LVX←X1K=XTK
[151]  LVY←Y2K=YTK
[152] ⍝
[153]  THETA1K←((X1K<XTK)×○1)+¯3○(YTK÷(X1K-XTK+LVX))
[154]  THETA2K←((Y2K<YTK)×○1)+¯3○(XTK÷(Y2K-YTK+LVY))
[155] ⍝
[156]  THETA1K←(LVX×0.5×○1)+(~LVX)×THETA1K
[157]  THETA2K←(LVY×0.5×○1)+(~LVY)×THETA2K
[158] ⍝
[159] ⍝CONTINUE WITH CALCULATIONS:
[160] ⍝
[161] ⍝DATAPOINTS←(2,NRP)⍴0
[162]  RNGPCERRSTORE←MSAVE←0
[163]  RESIDERRSTORE←NZ←2 1⍴0
[164]  GVSTORE←VERRSTORE←ERRSTORE←BSTORE←GSTORE←NZFIX←(2,K+1)⍴0
[165] ⍝
[166]  XHATSTORE←4 1⍴0
[167]  PSTORE←4 4⍴0
[168] ⍝
[169]  H←2 4⍴1 0 0 0 0 0 1 0
[170]  PHI←(⍳4)∘.=⍳4
[171]  PHI[1;2]←DT
[172]  PHI[3;4]←DT
[173] ⍝ACCEL←0.5×DT*2
[174] ⍝GAMMA←4 2⍴ACCEL,0,DT,0,0,ACCEL,0,DT
[175]  I←(⍳4)∘.=⍳4
[176] ⍝
[177] ⍝*****************************************************************
[178] ⍝
[179] ⍝ START MAIN LOOP:
[180] ⍝
[181]  N←1
[182] ⍝
[183] ⍝
[184]  LOOP:G←P+.×(⍉H)+.×⌹R+H+.×P+.×⍉H
[185] ⍝
[186] ⍝
[187] ⍝ 'KALMAN GAIN:'
[188] ⍝
[189]  GVSTORE[1;N]←G[2;1]
[190]  GVSTORE[2;N]←G[4;2]
[191]  GSTORE[1;N]←G[1;1]
[192]  GSTORE[2;N]←G[3;2]
[193] ⍝G
[194] ⍝
[195] ⍝
[196]  P←(I-G+.×H)+.×P
[197]  PSTORE←PSTORE,[0.5×1+1<N] P
[198] ⍝
[199] ⍝
[200] ⍝'UPDATED P:'
```

```
[201] ⍝P
[202]  ZK←2 1⍴XTK[N],YTK[N]
[203] ⍝'TRUE TARGET POSITION'
[204] ⍝ZK
[205] ⍝
[206] ⍝ THE NEXT SECTION GENERATES 2 NORMAL BEARING ERRORS.
[207] ⍝
[208]  U←(?2⍴10*10)÷10*10
[209]  SSQ←(+/2×⍟U)*0.5
[210]  SPR1←A1÷(R2D×1.96)
[211]  SPR2←A2÷(R2D×1.96)
[212]  BSTORE[1;N]←BE1←SPR1×SSQ×2○(U[2]×2×○1)
[213]  BSTORE[2;N]←BE2←SPR2×SSQ×1○(U[2]×2×○1)
[214] ⍝
[215]  NTHETA1←THETA1K[N]+BE1
[216]  NTHETA2←THETA2K[N]+BE2
[217] ⍝
[218]  SA1←1○NTHETA1
[219]  CA1←2○NTHETA1
[220]  SA2←1○NTHETA2
[221]  CA2←2○NTHETA2
[222]  D←2○(NTHETA1+NTHETA2)
[223] ⍝
[224]  NZFIX[1;N]←NZ[1;]←((Y2K[N]×SA2×CA1)-(X1K[N]×SA1×SA2))÷D
[225]  NZFIX[2;N]←NZ[2;]←((X1K[N]×SA1×CA2)-(Y2K[N]×SA2×SA1))÷D
[226] ⍝NOISE[1;N]←XTK[N]-NZFIX[1;N]
[227] ⍝NOISE[2;N]←YTK[N]-NZFIX[2;N]
[228] ⍝ 'TRUE TGT X,Y POSIT MINUS NOISY X,Y POSIT'
[229] ⍝ NOISE[;N]
[230] ⍝
[231] ⍝
[232] ⍝
[233]  RESIDERR←NZ-H+.×XHAT
[234]  XHAT←XHAT+G+.×RESIDERR
[235] ⍝
[236] ⍝
[237]  XHATSTORE←XHATSTORE,XHAT
[238] ⍝
[239]  RESIDERRSTORE←RESIDERRSTORE,RESIDERR
[240] ⍝
[241] ⍝DATAPOINTS←DATAPOINTS,NZ
[242] ⍝DATAPOINTS←DATAPOINTS,1 0 1 0⌿XHAT
[243] ⍝
[244] ⍝DATAUSE←(2,-NRP)↑DATAPOINTS
[245] ⍝SUMX←+/DATAUSE[1;]
[246] ⍝SUMY←+/DATAUSE[2;]
[247] ⍝SUMX2←+/DATAUSE[1;]*2
[248] ⍝SUMXY←+/DATAUSE[1;]×DATAUSE[2;]
[249] ⍝DENOMINATOR←(NRP×SUMX2)-SUMX*2
[250] ⍝M←((NRP×SUMXY)-SUMX×SUMY)÷DENOMINATOR
[251] ⍝MDEG←R2D×(○1)÷3○M
[252] ⍝MSAVE←MSAVE,MDEG
[253] ⍝
[254]  VERRSTORE[1;N]←VX[1+N>TURN+1]-XHAT[2;]
[255]  VERRSTORE[2;N]←VY[1+N>TURN+1]-XHAT[4;]
[256]  RTRUE←(+⌿ZK×ZK)*0.5
[257]  RX←1 0 1 0⌿XHAT
[258]  RHAT←(+⌿RX×RX)*0.5
[259]  RNGPCERR←100×(÷RTRUE)×|RTRUE-RHAT
[260]  RNGPCERRSTORE←RNGPCERRSTORE,RNGPCERR
[261] ⍝
[262] ⍝
[263] ⍝ 'UPDATED XHAT:'
[264] ⍝XHAT
[265]  ERR←ZK-2 1⍴,XHAT[1 3;]
[266] ⍝ 'ERROR VECTOR:'
[267]  ERRSTORE[1;N]←ERR[1;]
[268]  ERRSTORE[2;N]←ERR[2;]
[269] ⍝ERR
[270] ⍝
```

```
[271] ⍝
[272] P←Q+PHI+.×P+.×⍉PHI
[273] ⍝
[274] ⍝
[275] ⍝ 'PREDICTED P:'
[276] ⍝ P
[277] ⍝
[278] ⍝
[279] XHAT←PHI+.×XHAT
[280] ⍝
[281] ⍝
[282] ⍝ 'PREDICTED XHAT:'
[283] ⍝ XHAT
[284] ⍝
[285] ⍝X2←XHAT[2;]
[286] ⍝X4←XHAT[4;]
[287] ⍝XA←X4,X4,X2,X2,X4,X4,X2,X2,X2,X2,X2,X2,X2,X2,X2,X2
[288] ⍝XB←X4,X4,X4,X4,X4,X4,X4,X4,X4,X4,X2,X2,X4,X4,X2,X2
[289] ⍝QTERMS←C×F1×XA×XB×DTS*F4
[290] ⍝Q←4 4⍴QTERMS
[291] ⍝ CALCULATE TERMS IN THE R MATRIX:
[292] ⍝
[293] R1←(XHAT[3;]*2)+(XHAT[1;]-X1K[N])*2
[294] SIG12←R1×(3○A1÷R2D×1.96)*2
[295] R2←(XHAT[1;]*2)+(XHAT[3;]-Y2K[N])*2
[296] SIG22←R2×(3○A2÷R2D×1.96)*2
[297] ⍝
[298] D1←D*2
[299] UL←((SIG12×SA2*2)+(SIG22×CA1*2))÷D1
[300] LR←((SIG12×CA2*2)+(SIG22×SA1*2))÷D1
[301] CROSS←-((SIG12×SA2×CA2)+(SIG22×SA1×CA1))÷D1
[302] R←2 2⍴UL,CROSS,CROSS,LR
[303] ⍝ 'R MATRIX:'
[304] ⍝ R
[305] ⍝ ' '
[306] ⍝ ' ****************************************'
[307] ⍝ ' ITERATION NO. ',⍕(N+1)
[308] ⍝ ' '
[309] N←N+1
[310] →(N<K+1)/LOOP
[311] ⍝
[312] ⍝*************************************************************
[313] ⍝
[314] LQM←0.5×1+1<MAINLOOP
[315] QGVSTORE←QGVSTORE,[LQM] GVSTORE
[316] QGSTORE←QGSTORE,[LQM] GSTORE
[317] QVERRSTORE←QVERRSTORE,[LQM] VERRSTORE
[318] QERRSTORE←QERRSTORE,[LQM] ERRSTORE
[319] QRESIDERRSTORE←QRESIDERRSTORE,[LQM] RESIDERRSTORE
[320] RNGPCERRSTORE←(1,⍴RNGPCERRSTORE)⍴RNGPCERRSTORE
[321] QRNGPCERRSTORE←QRNGPCERRSTORE,[LQM] RNGPCERRSTORE
[322] ⍝MSAVE←(1,⍴MSAVE)⍴MSAVE
[323] ⍝QMSAVE←QMSAVE,[LQM] MSAVE
[324] MAINLOOP←MAINLOOP+1
[325] ⎕RL←RLSAVE
[326] →(MAINLOOP≤0.5×⍴B6)/TOP
[327] ' COMPLETE.'
```

LIST OF REFERENCES

1. Gelb, Arthur, *Applied Optimal Estimation*, The M.I.T. Press, Cambridge, MA., 1984.

2. Sorenson, H.W., *Kalman Filtering: Theory and Application*, IEEE Press, New York, NY, 1985.

3. Washburn, A., "A Short Introduction to Kalman Filters", Naval Postgraduate School, Monterey, CA (Unpublished Notes).

4. Kirk, D.E., "Optimal Estimation: An Introduction to the Theory and Applications", Naval Postgraduate School, Monterey, CA, 1975, (unpublished).

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Dr. H. A. Titus Department of Electrical and Computer Engineering, Code 32 Naval Postgraduate School, Monterey, CA 93943-5000	4
4.	LT Jay A. Gutzler 1513 Freeman Ct. Arlington, TX 76013	1